Cisco ASA Services Module CLI Configuration Guide

Software Version 8.5

Released: July 8, 2011
Updated: October 31, 2012

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

Text Part Number: N/A, Online only
THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL
STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT
WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB’s public
domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED “AS IS” WITH
ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT
LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF
DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING,
WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO
OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this
URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership
relationship between Cisco and any other company. (1110R)

Cisco ASA Services Module CLI Configuration Guide
Copyright © 2011-2012 Cisco Systems, Inc. All rights reserved.
## CONTENTS

### Part 1 Getting Started with the ASA

#### Chapter 1 Introduction to the Cisco ASA Services Module 1-1

- **Hardware and Software Compatibility** 1-1
- **New Features** 1-1
  - New Features in Version 8.5(1.7) 1-2
  - New Features in Version 8.5(1.6) 1-2
  - New Features in Version 8.5(1) 1-3
- **How the ASA Services Module Works with the Switch** 1-5
- **Firewall Functional Overview** 1-7
  - **Security Policy Overview** 1-8
    - Permitting or Denying Traffic with Access Lists 1-8
    - Applying NAT 1-8
    - Protecting from IP Fragments 1-8
    - Using AAA for Through Traffic 1-9
    - Applying HTTP, HTTPS, or FTP Filtering 1-9
    - Applying Application Inspection 1-9
    - Applying QoS Policies 1-9
    - Applying Connection Limits and TCP Normalization 1-9
    - Enabling Threat Detection 1-9
    - Enabling the Botnet Traffic Filter 1-10
- **Firewall Mode Overview** 1-10
  - **Stateful Inspection Overview** 1-10
- **Security Context Overview** 1-11

#### Chapter 2 Configuring the Switch for Use with the ASA Services Module 2-1

- **Information About the Switch** 2-1
Guidelines and Limitations 2-2
Verifying the Module Installation 2-3
Assigning VLANs to the ASA Services Module 2-4
Using the MSFC as a Directly Connected Router 2-5
  Information About SVIs 2-6
  Configuring SVIs 2-8
Configuring the Switch for ASA Failover 2-9
  Assigning VLANs to the Secondary ASA Services Module 2-10
  Adding a Trunk Between a Primary Switch and Secondary Switch 2-10
  Ensuring Compatibility with Transparent Firewall Mode 2-10
  Enabling Autostate Messaging for Rapid Link Failure Detection 2-10
Resetting the ASA Services Module 2-11
Monitoring the ASA Services Module 2-11
Feature History for the Switch for Use with the ASA Services Module 2-13

CHAPTER 3

Getting Started 3-1
Accessing the ASA Services Module Command-Line Interface 3-1
  Logging Into the ASA Services Module 3-1
    Information About Connection Methods 3-2
    Logging In 3-2
  Logging Out of a Console Session 3-3
    Logging Out 3-4
    Killing an Active Console Connection 3-4
  Logging Out of a Telnet Session 3-5
Configuring ASDM Access for the ASA Services Module 3-5
Starting ASDM 3-7
  Connecting to ASDM for the First Time 3-8
  Starting ASDM from the ASDM-IDM Launcher 3-9
  Starting ASDM from the Java Web Start Application 3-9
  Using ASDM in Demo Mode 3-10
Working with the Configuration 3-11
  Saving Configuration Changes 3-12
    Saving Configuration Changes in Single Context Mode 3-12
    Saving Configuration Changes in Multiple Context Mode 3-12
  Copying the Startup Configuration to the Running Configuration 3-14
  Viewing the Configuration 3-14
  Clearing and Removing Configuration Settings 3-15
  Creating Text Configuration Files Offline 3-15
Chapter 4: Managing Feature Licenses

Supported Feature Licenses
- Licenses
- License Notes

Information About Feature Licenses
- Preinstalled License
- Permanent License
- Time-Based Licenses
  - Time-Based License Activation Guidelines
  - How the Time-Based License Timer Works
  - How Permanent and Time-Based Licenses Combine
  - Stacking Time-Based Licenses
  - Time-Based License Expiration

Failover Licenses
- Failover License Requirements and Exceptions
- How Failover Licenses Combine
- Loss of Communication Between Failover Units
- Upgrading Failover Pairs
- No Payload Encryption Models
- Licenses FAQ

Guidelines and Limitations

Configuring Licenses
- Obtaining an Activation Key
- Activating or Deactivating Keys

Monitoring Licenses

Feature History for Licensing

Part 2: Configuring Firewall and Security Context Modes

Chapter 5: Configuring the Transparent or Routed Firewall

Configuring the Firewall Mode
- Information About the Firewall Mode
- Information About Routed Firewall Mode
- Information About Transparent Firewall Mode

Licensing Requirements for the Firewall Mode

Default Settings

Guidelines and Limitations
Contents

Setting the Firewall Mode 5-8
Feature History for Firewall Mode 5-9
Configuring ARP Inspection for the Transparent Firewall 5-9
  Information About ARP Inspection 5-10
  Licensing Requirements for ARP Inspection 5-10
  Default Settings 5-10
  Guidelines and Limitations 5-11
  Configuring ARP Inspection 5-11
    Task Flow for Configuring ARP Inspection 5-11
    Adding a Static ARP Entry 5-11
    Enabling ARP Inspection 5-12
    Monitoring ARP Inspection 5-13
Feature History for ARP Inspection 5-13
Customizing the MAC Address Table for the Transparent Firewall 5-13
  Information About the MAC Address Table 5-13
  Licensing Requirements for the MAC Address Table 5-14
  Default Settings 5-14
  Guidelines and Limitations 5-14
  Configuring the MAC Address Table 5-15
    Adding a Static MAC Address 5-15
    Setting the MAC Address Timeout 5-15
    Disabling MAC Address Learning 5-16
  Monitoring the MAC Address Table 5-16
Feature History for the MAC Address Table 5-17
Firewall Mode Examples 5-17
  How Data Moves Through the ASA in Routed Firewall Mode 5-17
    An Inside User Visits a Web Server 5-18
    An Outside User Visits a Web Server on the DMZ 5-19
    An Inside User Visits a Web Server on the DMZ 5-20
    An Outside User Attempts to Access an Inside Host 5-21
    A DMZ User Attempts to Access an Inside Host 5-22
  How Data Moves Through the Transparent Firewall 5-23
    An Inside User Visits a Web Server 5-24
    An Inside User Visits a Web Server Using NAT 5-25
    An Outside User Visits a Web Server on the Inside Network 5-26
    An Outside User Attempts to Access an Inside Host 5-27

CHAPTER 6
Configuring Multiple Context Mode 6-1
  Information About Security Contexts 6-1
Common Uses for Security Contexts  6-2
Context Configuration Files  6-2
  Context Configurations  6-2
  System Configuration  6-2
  Admin Context Configuration  6-2
How the ASA Classifies Packets  6-3
  Valid Classifier Criteria  6-3
  Classification Examples  6-4
Cascading Security Contexts  6-6
Management Access to Security Contexts  6-7
  System Administrator Access  6-7
  Context Administrator Access  6-8
Information About Resource Management  6-8
  Resource Limits  6-8
  Default Class  6-9
  Class Members  6-10
Information About MAC Addresses  6-11
  Default MAC Address  6-11
  Interaction with Manual MAC Addresses  6-11
  Failover MAC Addresses  6-12
  MAC Address Format  6-12
Licensing Requirements for Multiple Context Mode  6-12
Guidelines and Limitations  6-13
Default Settings  6-13
Configuring Multiple Contexts  6-14
  Task Flow for Configuring Multiple Context Mode  6-14
  Enabling or Disabling Multiple Context Mode  6-14
    Enabling Multiple Context Mode  6-14
    Restoring Single Context Mode  6-15
  Configuring a Class for Resource Management  6-15
  Configuring a Security Context  6-17
    Automatically Assigning MAC Addresses to Context Interfaces  6-21
Changing Between Contexts and the System Execution Space  6-22
Managing Security Contexts  6-22
  Removing a Security Context  6-22
  Changing the Admin Context  6-23
  Changing the Security Context URL  6-23
  Reloading a Security Context  6-25
    Reloading by Clearing the Configuration  6-25
Reloading by Removing and Re-adding the Context 6-25
Monitoring Security Contexts 6-26
  Viewing Context Information 6-26
  Viewing Resource Allocation 6-27
  Viewing Resource Usage 6-30
  Monitoring SYN Attacks in Contexts 6-31
  Viewing Assigned MAC Addresses 6-33
    Viewing MAC Addresses in the System Configuration 6-34
    Viewing MAC Addresses Within a Context 6-35
Configuration Examples for Multiple Context Mode 6-36
Feature History for Multiple Context Mode 6-37

PART 3
Configuring Interfaces

CHAPTER 7
Configuring Interfaces (Routed Mode) 7-1
  Information About Completing Interface Configuration in Routed Mode 7-1
    Security Levels 7-1
    Dual IP Stack (IPv4 and IPv6) 7-2
  Licensing Requirements for Completing Interface Configuration in Routed Mode 7-2
  Guidelines and Limitations 7-3
  Default Settings 7-3
  Completing Interface Configuration in Routed Mode 7-4
    Task Flow for Completing Interface Configuration 7-4
    Configuring General Interface Parameters 7-4
    Configuring the MAC Address and MTU 7-6
    Configuring IPv6 Addressing 7-8
      Information About IPv6 7-8
        Configuring a Global IPv6 Address and Other Options 7-10
        Allowing Same Security Level Communication 7-12
  Turning Off and Turning On Interfaces 7-14
  Monitoring Interfaces 7-14
  Feature History for Interfaces in Routed Mode 7-15

CHAPTER 8
Configuring Interfaces (Transparent Mode) 8-1
  Information About Completing Interface Configuration in Transparent Mode 8-1
  Bridge Groups in Transparent Mode 8-1
  Security Levels 8-2
  Licensing Requirements for Completing Interface Configuration in Transparent Mode 8-2
Guidelines and Limitations

Default Settings

Completing Interface Configuration in Transparent Mode
  Task Flow for Completing Interface Configuration
  Configuring Bridge Groups
  Configuring General Interface Parameters
  Configuring the MAC Address and MTU
  Configuring IPv6 Addressing
    Information About IPv6
    Configuring a Global IPv6 Address and Other Options
  Allowing Same Security Level Communication

Turning Off and Turning On Interfaces

Monitoring Interfaces

Configuration Examples for Interfaces in Transparent Mode

Feature History for Interfaces in Transparent Mode

PART 4
Configuring Basic Settings

CHAPTER 9
Configuring Basic Settings
  Configuring the Hostname, Domain Name, and Passwords
    Changing the Login Password
    Changing the Enable Password
    Setting the Hostname
    Setting the Domain Name
  Configuring the Master Passphrase
    Information About the Master Passphrase
    Licensing Requirements for the Master Passphrase
    Guidelines and Limitations
    Adding or Changing the Master Passphrase
    Disabling the Master Passphrase
    Recovering the Master Passphrase
    Feature History for the Master Passphrase
  Configuring the DNS Server
  Monitoring DNS Cache
    DNS Cache Monitoring Commands
  Feature History for DNS Cache
CHAPTER 10
Configuring DHCP 10-1
  Information About DHCP 10-1
  Licensing Requirements for DHCP 10-1
  Guidelines and Limitations 10-2
  Configuring a DHCP Server 10-2
    Enabling the DHCP Server 10-3
    Configuring DHCP Options 10-4
      Options that Return an IP Address 10-4
      Options that Return a Text String 10-4
      Options that Return a Hexadecimal Value 10-5
    Using Cisco IP Phones with a DHCP Server 10-6
  Configuring DHCP Relay Services 10-7
  DHCP Monitoring Commands 10-8
  Feature History for DHCP 10-8

CHAPTER 11
Configuring Dynamic DNS 11-1
  Information About DDNS 11-1
  Licensing Requirements for DDNS 11-2
  Guidelines and Limitations 11-2
  Configuring DDNS 11-2
  Configuration Examples for DDNS 11-3
    Example 1: Client Updates Both A and PTR RRs for Static IP Addresses 11-3
    Example 2: Client Updates Both A and PTR RRs; DHCP Server Honors Client Update Request; FQDN Provided Through Configuration 11-3
    Example 3: Client Includes FQDN Option Instructing Server Not to Update Either RR; Server Overrides Client and Updates Both RRs. 11-4
    Example 4: Client Asks Server To Perform Both Updates; Server Configured to Update PTR RR Only; Honors Client Request and Updates Both A and PTR RR 11-5
    Example 5: Client Updates A RR; Server Updates PTR RR 11-5
  DDNS Monitoring Commands 11-6
  Feature History for DDNS 11-6

PART 5
Configuring Objects and Access Lists

CHAPTER 12
Configuring Objects 12-1
  Configuring Objects and Groups 12-1
  Information About Objects and Groups 12-1
  Information About Objects 12-2
Information About Object Groups 12-2
Licensing Requirements for Objects and Groups 12-2
Guidelines and Limitations for Objects and Groups 12-3
Configuring Objects 12-3
  Configuring a Network Object 12-3
  Configuring a Service Object 12-4
Configuring Object Groups 12-6
  Adding a Protocol Object Group 12-6
  Adding a Network Object Group 12-7
  Adding a Service Object Group 12-8
  Adding an ICMP Type Object Group 12-9
Nesting Object Groups 12-10
Removing Object Groups 12-11
Monitoring Objects and Groups 12-11
Feature History for Objects and Groups 12-12
Configuring Regular Expressions 12-12
  Creating a Regular Expression 12-12
  Creating a Regular Expression Class Map 12-15
Scheduling Extended Access List Activation 12-16
  Information About Scheduling Access List Activation 12-16
  Licensing Requirements for Scheduling Access List Activation 12-16
  Guidelines and Limitations for Scheduling Access List Activation 12-16
Configuring and Applying Time Ranges 12-17
Configuration Examples for Scheduling Access List Activation 12-18
Feature History for Scheduling Access List Activation 12-18

CHAPTER 13
Information About Access Lists 13-1
  Access List Types 13-1
  Access Control Entry Order 13-2
  Access Control Implicit Deny 13-3
  IP Addresses Used for Access Lists When You Use NAT 13-3
  Where to Go Next 13-3

CHAPTER 14
Adding an Extended Access List 14-1
  Information About Extended Access Lists 14-1
  Licensing Requirements for Extended Access Lists 14-1
  Default Settings 14-2
  Configuring Extended Access Lists 14-2
Adding an Extended Access List 14-3
Adding Remarks to Access Lists 14-5
Monitoring Extended Access Lists 14-5
Configuration Examples for Extended Access Lists 14-5
  Configuration Examples for Extended Access Lists (No Objects) 14-6
  Configuration Examples for Extended Access Lists (Using Objects) 14-6
Where to Go Next 14-7
Feature History for Extended Access Lists 14-7

Chapter 15

Adding an EtherType Access List 15-1
  Information About EtherType Access Lists 15-1
  Licensing Requirements for EtherType Access Lists 15-1
  Guidelines and Limitations 15-2
  Default Settings 15-2
  Configuring EtherType Access Lists 15-2
    Task Flow for Configuring EtherType Access Lists 15-2
    Adding EtherType Access Lists 15-3
    Adding Remarks to Access Lists 15-4
  What to Do Next 15-4
  Monitoring EtherType Access Lists 15-4
  Configuration Examples for EtherType Access Lists 15-5
  Feature History for EtherType Access Lists 15-5

Chapter 16

Adding a Standard Access List 16-1
  Information About Standard Access Lists 16-1
  Licensing Requirements for Standard Access Lists 16-1
  Guidelines and Limitations 16-1
  Default Settings 16-2
  Adding Standard Access Lists 16-3
    Task Flow for Configuring Extended Access Lists 16-3
    Adding a Standard Access List 16-3
    Adding Remarks to Access Lists 16-4
  What to Do Next 16-4
  Monitoring Access Lists 16-4
  Configuration Examples for Standard Access Lists 16-4
  Feature History for Standard Access Lists 16-5
CHAPTER 17

Adding an IPv6 Access List 17-1
  Information About IPv6 Access Lists 17-1
  Licensing Requirements for IPv6 Access Lists 17-1
  Prerequisites for Adding IPv6 Access Lists 17-2
  Guidelines and Limitations 17-2
  Default Settings 17-3
  Configuring IPv6 Access Lists 17-4
    Task Flow for Configuring IPv6 Access Lists 17-4
    Adding IPv6 Access Lists 17-5
    Adding Remarks to Access Lists 17-6
  Monitoring IPv6 Access Lists 17-7
  Configuration Examples for IPv6 Access Lists 17-7
  Where to Go Next 17-7
  Feature History for IPv6 Access Lists 17-7

CHAPTER 18

Configuring Logging for Access Lists 18-1
  Configuring Logging for Access Lists 18-1
    Information About Logging Access List Activity 18-1
    Licensing Requirements for Access List Logging 18-2
    Guidelines and Limitations 18-2
    Default Settings 18-3
    Configuring Access List Logging 18-3
    Monitoring Access Lists 18-4
    Configuration Examples for Access List Logging 18-4
    Feature History for Access List Logging 18-5
  Managing Deny Flows 18-5
    Information About Managing Deny Flows 18-6
    Licensing Requirements for Managing Deny Flows 18-6
    Guidelines and Limitations 18-6
    Default Settings 18-7
    Managing Deny Flows 18-7
    Monitoring Deny Flows 18-7
    Feature History for Managing Deny Flows 18-8

PART 6

Configuring IP Routing

CHAPTER 19

Routing Overview 19-1
  Information About Routing 19-1
Switching 19-2  
Path Determination 19-2  
Supported Route Types 19-2  
  Static Versus Dynamic 19-3  
  Single-Path Versus Multipath 19-3  
  Flat Versus Hierarchical 19-3  
  Link-State Versus Distance Vector 19-4  
How Routing Behaves Within the ASA 19-4  
Egress Interface Selection Process 19-4  
Next Hop Selection Process 19-4  
Supported Internet Protocols for Routing 19-5  
Information About the Routing Table 19-6  
  Displaying the Routing Table 19-6  
  How the Routing Table Is Populated 19-6  
  Backup Routes 19-8  
  How Forwarding Decisions Are Made 19-8  
  Dynamic Routing and Failover 19-9  
Information About IPv6 Support 19-9  
  Features That Support IPv6 19-9  
  IPv6-Enabled Commands 19-10  
  Entering IPv6 Addresses in Commands 19-11  
Disabling Proxy ARPs 19-11  

**CHAPTER 20**  
Configuring Static and Default Routes 20-1  
  Information About Static and Default Routes 20-1  
  Licensing Requirements for Static and Default Routes 20-2  
  Guidelines and Limitations 20-2  
  Configuring Static and Default Routes 20-2  
    Configuring a Static Route 20-3  
      Adding or Editing a Static Route 20-3  
    Configuring a Default Static Route 20-4  
      Limitations on Configuring a Default Static Route 20-4  
    Configuring IPv6 Default and Static Routes 20-5  
  Monitoring a Static or Default Route 20-6  
  Configuration Examples for Static or Default Routes 20-8  
  Feature History for Static and Default Routes 20-8
CHAPTER 23
RIP Stability Features 23-2
RIP Timers 23-2
Licensing Requirements for RIP 23-3
Guidelines and Limitations 23-3
Configuring RIP 23-4
   Enabling RIP 23-4
Customizing RIP 23-4
   Configuring the RIP Version 23-5
   Configuring Interfaces for RIP 23-6
   Configuring the RIP Send and Receive Version on an Interface 23-6
   Configuring Route Summarization 23-7
   Filtering Networks in RIP 23-8
   Redistributing Routes into the RIP Routing Process 23-8
   Enabling RIP Authentication 23-9
   Restarting the RIP Process 23-10
Monitoring RIP 23-11
Configuration Example for RIP 23-11
Feature History for RIP 23-11

CHAPTER 24
Configuring Multicast Routing 24-1
Information About Multicast Routing 24-1
   Stub Multicast Routing 24-2
   PIM Multicast Routing 24-2
   Multicast Group Concept 24-2
   Multicast Addresses 24-2
Licensing Requirements for Multicast Routing 24-2
Guidelines and Limitations 24-3
Enabling Multicast Routing 24-3
Customizing Multicast Routing 24-4
   Configuring Stub Multicast Routing and Forwarding IGMP Messages 24-4
   Configuring a Static Multicast Route 24-4
   Configuring IGMP Features 24-5
      Disabling IGMP on an Interface 24-6
      Configuring IGMP Group Membership 24-6
      Configuring a Statically Joined IGMP Group 24-6
      Controlling Access to Multicast Groups 24-7
      Limiting the Number of IGMP States on an Interface 24-7
      Modifying the Query Messages to Multicast Groups 24-8
Configuration Example for EIGRP  25-18
Feature History for EIGRP  25-19

CHAPTER 26

Configuring IPv6 Neighbor Discovery  26-1
Information About IPv6 Neighbor Discovery  26-1
Neighbor Solicitation Messages  26-2
Neighbor Reachable Time  26-3
Router Advertisement Messages  26-3
Static IPv6 Neighbors  26-4
Licensing Requirements for IPv6 Neighbor Discovery  26-4
Guidelines and Limitations  26-4
Default Settings for IPv6 Neighbor Discovery  26-6
Configuring the Neighbor Solicitation Message Interval  26-7
Configuring the Neighbor Reachable Time  26-7
Configuring the Router Advertisement Transmission Interval  26-8
Configuring the Router Lifetime Value  26-8
Configuring DAD Settings  26-9
Configuring IPv6 Addresses on an Interface  26-9
Suppressing Router Advertisement Messages  26-10
Configuring the IPv6 Prefix  26-11
Configuring a Static IPv6 Neighbor  26-12
Monitoring IPv6 Neighbor Discovery  26-13
Additional References  26-13
  Related Documents for IPv6 Prefixes  26-14
  RFCs for IPv6 Prefixes and Documentation  26-14
Feature History for IPv6 Neighbor Discovery  26-14

PART 7

Configuring Network Address Translation

CHAPTER 27

Information About NAT  27-1
Why Use NAT?  27-1
NAT Terminology  27-2
NAT Types  27-3
  NAT Types Overview  27-3
Static NAT  27-3
    Information About Static NAT  27-3
    Information About Static NAT with Port Translation  27-4
Information About One-to-Many Static NAT 27-6
Information About Other Mapping Scenarios (Not Recommended) 27-7
Dynamic NAT 27-8
Information About Dynamic NAT 27-9
Dynamic NAT Disadvantages and Advantages 27-10
Dynamic PAT 27-10
Information About Dynamic PAT 27-10
Dynamic PAT Disadvantages and Advantages 27-11
Identity NAT 27-11
NAT in Routed and Transparent Mode 27-12
NAT in Routed Mode 27-13
NAT in Transparent Mode 27-13
NAT for VPN 27-14
How NAT is Implemented 27-16
Main Differences Between Network Object NAT and Twice NAT 27-16
Information About Network Object NAT 27-17
Information About Twice NAT 27-17
NAT Rule Order 27-20
NAT Interfaces 27-21
Routing NAT Packets 27-21
Mapped Addresses and Routing 27-22
Transparent Mode Routing Requirements for Remote Networks 27-23
Determining the Egress Interface 27-24
DNS and NAT 27-24
Where to Go Next 27-27

CHAPTER 28
Configuring Network Object NAT 28-1
Information About Network Object NAT 28-1
Licensing Requirements for Network Object NAT 28-2
Prerequisites for Network Object NAT 28-2
Guidelines and Limitations 28-2
Default Settings 28-3
Configuring Network Object NAT 28-3
Configuring Dynamic NAT 28-4
Configuring Dynamic PAT (Hide) 28-6
Configuring Static NAT or Static NAT-with-Port-Translation 28-9
Configuring Identity NAT 28-12
Monitoring Network Object NAT 28-14
Configuration Examples for Network Object NAT 28-15
Providing Access to an Inside Web Server (Static NAT) 28-15
NAT for Inside Hosts (Dynamic NAT) and NAT for an Outside Web Server (Static NAT) 28-16
Inside Load Balancer with Multiple Mapped Addresses (Static NAT, One-to-Many) 28-17
Single Address for FTP, HTTP, and SMTP (Static NAT-with-Port-Translation) 28-18
DNS Server on Mapped Interface, Web Server on Real Interface (Static NAT with DNS Modification) 28-19
DNS Server and Web Server on Mapped Interface, Web Server is Translated (Static NAT with DNS Modification) 28-21
Feature History for Network Object NAT 28-22

CHAPTER 29
Configuring Twice NAT 29-1
Information About Twice NAT 29-1
Licensing Requirements for Twice NAT 29-2
Prerequisites for Twice NAT 29-2
Guidelines and Limitations 29-2
Default Settings 29-3
Configuring Twice NAT 29-3
Configuring Dynamic NAT 29-4
Configuring Dynamic PAT (Hide) 29-8
Configuring Static NAT or Static NAT-with-Port-Translation 29-13
Configuring Identity NAT 29-18
Monitoring Twice NAT 29-22
Configuration Examples for Twice NAT 29-22
Different Translation Depending on the Destination (Dynamic PAT) 29-22
Different Translation Depending on the Destination Address and Port (Dynamic PAT) 29-24
Feature History for Twice NAT 29-26

PART 8
Configuring Service Policies Using the Modular Policy Framework

CHAPTER 30
Configuring a Service Policy Using the Modular Policy Framework 30-1
Information About Service Policies 30-1
Supported Features for Through Traffic 30-2
Supported Features for Management Traffic 30-2
Feature Directionality 30-2
Feature Matching Within a Service Policy 30-3
Order in Which Multiple Feature Actions are Applied 30-4
Incompatibility of Certain Feature Actions 30-4
Contents

| Feature Matching for Multiple Service Policies | 30-5 |
| Licensing Requirements for Service Policies | 30-6 |
| Guidelines and Limitations | 30-6 |
| Default Settings | 30-7 |
| Default Configuration | 30-7 |
| Default Class Maps | 30-8 |
| Task Flows for Configuring Service Policies | 30-9 |
| Task Flow for Using the Modular Policy Framework | 30-9 |
| Identifying Traffic (Layer 3/4 Class Maps) | 30-12 |
| Creating a Layer 3/4 Class Map for Through Traffic | 30-12 |
| Creating a Layer 3/4 Class Map for Management Traffic | 30-14 |
| Defining Actions (Layer 3/4 Policy Map) | 30-15 |
| Applying Actions to an Interface (Service Policy) | 30-17 |
| Monitoring Modular Policy Framework | 30-18 |
| Configuration Examples for Modular Policy Framework | 30-18 |
| Applying Inspection and QoS Policing to HTTP Traffic | 30-19 |
| Applying Inspection to HTTP Traffic Globally | 30-19 |
| Applying Inspection and Connection Limits to HTTP Traffic to Specific Servers | 30-20 |
| Applying Inspection to HTTP Traffic with NAT | 30-21 |
| Feature History for Service Policies | 30-21 |

**CHAPTER 31**

Configuring Special Actions for Application Inspections (Inspection Policy Map) | 31-1

| Information About Inspection Policy Maps | 31-1 |
| Guidelines and Limitations | 31-2 |
| Default Inspection Policy Maps | 31-2 |
| Defining Actions in an Inspection Policy Map | 31-2 |
| Identifying Traffic in an Inspection Class Map | 31-6 |
| Where to Go Next | 31-7 |

**PART 9**

Configuring Access Control

**CHAPTER 32**

Configuring Access Rules | 32-1

| Information About Access Rules | 32-1 |
| General Information About Rules | 32-2 |
| Implicit Permits | 32-2 |
Using Access Rules and EtherType Rules on the Same Interface  32-2
Implicit Deny  32-3
Inbound and Outbound Rules  32-3
Information About Extended Access Rules  32-4
Access Rules for Returning Traffic  32-4
Allowing Broadcast and Multicast Traffic through the Transparent Firewall Using Access Rules  32-5
Management Access Rules  32-5
Information About EtherType Rules  32-5
Supported EtherTypes and Other Traffic  32-6
Access Rules for Returning Traffic  32-6
Allowing MPLS  32-6
Licensing Requirements for Access Rules  32-6
Prerequisites  32-6
Guidelines and Limitations  32-7
Default Settings  32-7
Configuring Access Rules  32-7
Monitoring Access Rules  32-8
Configuration Examples for Permitting or Denying Network Access  32-9
Feature History for Access Rules  32-10

CHAPTER 33
Configuring AAA Servers and the Local Database  33-1
Information About AAA  33-1
Information About Authentication  33-2
Information About Authorization  33-2
Information About Accounting  33-3
Summary of Server Support  33-3
RADIUS Server Support  33-3
Authentication Methods  33-4
Attribute Support  33-4
RADIUS Authorization Functions  33-4
TACACS+ Server Support  33-5
RSA/SDI Server Support  33-5
RSA/SDI Version Support  33-5
Two-step Authentication Process  33-5
RSA/SDI Primary and Replica Servers  33-6
NT Server Support  33-6
Kerberos Server Support  33-6
LDAP Server Support  33-6
Authentication with LDAP  33-6
LDAP Server Types  33-7
Local Database Support, Including as a Fallback Method  33-7
How Fallback Works with Multiple Servers in a Group  33-8
Using Certificates and User Login Credentials  33-8
  Using User Login Credentials  33-8
  Using Certificates  33-9
Licensing Requirements for AAA Servers  33-9
Guidelines and Limitations  33-9
Configuring AAA  33-10
  Task Flow for Configuring AAA  33-10
  Configuring AAA Server Groups  33-11
  Configuring LDAP Attribute Maps  33-16
  Adding a User Account to the Local Database  33-19
    Guidelines  33-19
    Limitations  33-19
Managing User Passwords  33-23
  Changing User Passwords  33-24
Authenticating Users with a Public Key for SSH  33-25
Differentiating User Roles Using AAA  33-25
  Using Local Authentication  33-25
  Using RADIUS Authentication  33-26
  Using LDAP Authentication  33-26
  Using TACACS+ Authentication  33-27
Monitoring AAA Servers  33-27
Additional References  33-28
  RFCs  33-28
  Feature History for AAA Servers  33-28

CHAPTER 34  Configuring Management Access  34-1
Configuring ASA Access for ASDM, Telnet, or SSH  34-1
  Licensing Requirements for ASA Access for ASDM, Telnet, or SSH  34-1
  Guidelines and Limitations  34-2
  Configuring Telnet Access  34-3
  Using a Telnet Client  34-4
  Configuring SSH Access  34-4
  Using an SSH Client  34-5
  Configuring HTTPS Access for ASDM  34-6
Configuring CLI Parameters  34-6
CHAPTER 35
Configuring AAA Rules for Network Access    35-1
  AAA Performance    35-1
  Licensing Requirements for AAA Rules    35-1
  Guidelines and Limitations    35-2
  Configuring Authentication for Network Access    35-2
    Information About Authentication    35-2
    One-Time Authentication    35-2
    Applications Required to Receive an Authentication Challenge    35-2
    ASASM Authentication Prompts    35-3
    Static PAT and HTTP    35-4
  Configuring Network Access Authentication    35-4
  Enabling Secure Authentication of Web Clients    35-6
  Authenticating Directly with the ASASM    35-7
    Authenticating HTTP(S) Connections with a Virtual Server    35-8
    Authenticating Telnet Connections with a Virtual Server    35-9
  Configuring Authorization for Network Access    35-11
    Configuring TACACS+ Authorization    35-11
    Configuring RADIUS Authorization    35-14
      Configuring a RADIUS Server to Send Downloadable Access Control Lists    35-14
      Configuring a RADIUS Server to Download Per-User Access Control List Names    35-18
  Configuring Accounting for Network Access    35-18
    Using MAC Addresses to Exempt Traffic from Authentication and Authorization    35-20
    Feature History for AAA Rules    35-21

CHAPTER 36
Configuring Filtering Services    36-1
  Information About Web Traffic Filtering    36-1
  Configuring ActiveX Filtering    36-2
    Information About ActiveX Filtering    36-2
  Licensing Requirements for ActiveX Filtering    36-2
    Guidelines and Limitations for ActiveX Filtering    36-3
    Configuring ActiveX Filtering    36-3
    Configuration Examples for ActiveX Filtering    36-3
    Feature History for ActiveX Filtering    36-4
  Configuring Java Applet Filtering    36-4
    Information About Java Applet Filtering    36-4
    Licensing Requirements for Java Applet Filtering    36-4
    Guidelines and Limitations for Java Applet Filtering    36-5
    Configuring Java Applet Filtering    36-5
## Contents

- Configuration Examples for Java Applet Filtering 36-5
- Feature History for Java Applet Filtering 36-6
- Filtering URLs and FTP Requests with an External Server 36-6
- Information About URL Filtering 36-6
- Licensing Requirements for URL Filtering 36-7
- Guidelines and Limitations for URL Filtering 36-7
- Identifying the Filtering Server 36-8
- Configuring Additional URL Filtering Settings 36-10
  - Buffering the Content Server Response 36-10
  - Caching Server Addresses 36-11
  - Filtering HTTP URLs 36-11
  - Filtering HTTPS URLs 36-13
  - Filtering FTP Requests 36-14
- Monitoring Filtering Statistics 36-15
- Feature History for URL Filtering 36-17

### CHAPTER 37

Confusing Web Cache Services Using WCCP 37-1

- Information About WCCP 37-1
- Guidelines and Limitations 37-1
- Licensing Requirements for WCCP 37-2
- Enabling WCCP Redirection 37-3
- WCCP Monitoring Commands 37-4
- Feature History for WCCP 37-4

### CHAPTER 38

Configuring Digital Certificates 38-1

- Information About Digital Certificates 38-1
  - Public Key Cryptography 38-2
  - Certificate Scalability 38-2
  - Key Pairs 38-2
  - Trustpoints 38-3
    - Certificate Enrollment 38-3
    - Proxy for SCEP Requests 38-3
  - Revocation Checking 38-4
  - Supported CA Servers 38-4
  - CRLs 38-4
  - OCSP 38-5
  - The Local CA 38-6
  - Storage for Local CA Files 38-6
<table>
<thead>
<tr>
<th>Contents</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Local CA Server</td>
<td>38-6</td>
</tr>
<tr>
<td>Licensing Requirements for Digital Certificates</td>
<td>38-7</td>
</tr>
<tr>
<td>Prerequisites for Local Certificates</td>
<td>38-7</td>
</tr>
<tr>
<td>Prerequisites for SCEP Proxy Support</td>
<td>38-7</td>
</tr>
<tr>
<td>Guidelines and Limitations</td>
<td>38-8</td>
</tr>
<tr>
<td>Configuring Digital Certificates</td>
<td>38-9</td>
</tr>
<tr>
<td>Configuring Key Pairs</td>
<td>38-9</td>
</tr>
<tr>
<td>Removing Key Pairs</td>
<td>38-10</td>
</tr>
<tr>
<td>Configuring Trustpoints</td>
<td>38-10</td>
</tr>
<tr>
<td>Configuring CRLs for a Trustpoint</td>
<td>38-13</td>
</tr>
<tr>
<td>Exporting a Trustpoint Configuration</td>
<td>38-15</td>
</tr>
<tr>
<td>Importing a Trustpoint Configuration</td>
<td>38-16</td>
</tr>
<tr>
<td>Configuring CA Certificate Map Rules</td>
<td>38-17</td>
</tr>
<tr>
<td>Obtaining Certificates Manually</td>
<td>38-18</td>
</tr>
<tr>
<td>Obtaining Certificates Automatically with SCEP</td>
<td>38-20</td>
</tr>
<tr>
<td>Configuring Proxy Support for SCEP Requests</td>
<td>38-21</td>
</tr>
<tr>
<td>Enabling the Local CA Server</td>
<td>38-22</td>
</tr>
<tr>
<td>Configuring the Local CA Server</td>
<td>38-23</td>
</tr>
<tr>
<td>Customizing the Local CA Server</td>
<td>38-25</td>
</tr>
<tr>
<td>Debugging the Local CA Server</td>
<td>38-26</td>
</tr>
<tr>
<td>Disabling the Local CA Server</td>
<td>38-26</td>
</tr>
<tr>
<td>Deleting the Local CA Server</td>
<td>38-26</td>
</tr>
<tr>
<td>Configuring Local CA Certificate Characteristics</td>
<td>38-27</td>
</tr>
<tr>
<td>Configuring the Issuer Name</td>
<td>38-28</td>
</tr>
<tr>
<td>Configuring the CA Certificate Lifetime</td>
<td>38-28</td>
</tr>
<tr>
<td>Configuring the User Certificate Lifetime</td>
<td>38-29</td>
</tr>
<tr>
<td>Configuring the CRL Lifetime</td>
<td>38-30</td>
</tr>
<tr>
<td>Configuring the Server Keysize</td>
<td>38-30</td>
</tr>
<tr>
<td>Setting Up External Local CA File Storage</td>
<td>38-31</td>
</tr>
<tr>
<td>Downloading CRLs</td>
<td>38-33</td>
</tr>
<tr>
<td>Storing CRLs</td>
<td>38-34</td>
</tr>
<tr>
<td>Setting Up Enrollment Parameters</td>
<td>38-35</td>
</tr>
<tr>
<td>Adding and Enrolling Users</td>
<td>38-36</td>
</tr>
<tr>
<td>Renewing Users</td>
<td>38-38</td>
</tr>
<tr>
<td>Restoring Users</td>
<td>38-39</td>
</tr>
<tr>
<td>Removing Users</td>
<td>38-39</td>
</tr>
<tr>
<td>Revoking Certificates</td>
<td>38-40</td>
</tr>
<tr>
<td>Maintaining the Local CA Certificate Database</td>
<td>38-40</td>
</tr>
<tr>
<td>Rolling Over Local CA Certificates</td>
<td>38-40</td>
</tr>
</tbody>
</table>
Contents

Archiving the Local CA Server Certificate and Keypair 38-41
Monitoring Digital Certificates 38-41
Feature History for Certificate Management 38-43

PART 10

CHAPTER 39

Configuring Application Inspection

Getting Started with Application Layer Protocol Inspection 39-1
Information about Application Layer Protocol Inspection 39-1
How Inspection Engines Work 39-1
When to Use Application Protocol Inspection 39-2
Guidelines and Limitations 39-3
Default Settings 39-4
Configuring Application Layer Protocol Inspection 39-6

CHAPTER 40

Configuring Inspection of Basic Internet Protocols 40-1
DNS Inspection 40-1
How DNS Application Inspection Works 40-2
How DNS Rewrite Works 40-2
Configuring DNS Rewrite 40-3
Configuring DNS Rewrite with Two NAT Zones 40-4
Overview of DNS Rewrite with Three NAT Zones 40-4
Configuring DNS Rewrite with Three NAT Zones 40-6
Configuring a DNS Inspection Policy Map for Additional Inspection Control 40-7
Verifying and Monitoring DNS Inspection 40-10
FTP Inspection 40-11
FTP Inspection Overview 40-11
Using the strict Option 40-11
Configuring an FTP Inspection Policy Map for Additional Inspection Control 40-12
Verifying and Monitoring FTP Inspection 40-16
HTTP Inspection 40-16
HTTP Inspection Overview 40-16
Configuring an HTTP Inspection Policy Map for Additional Inspection Control 40-17
ICMP Inspection 40-20
ICMP Error Inspection 40-21
Instant Messaging Inspection 40-21
IM Inspection Overview 40-21
Configuring an Instant Messaging Inspection Policy Map for Additional Inspection Control 40-21
IP Options Inspection 40-24
Cisco ASA Services Module CLI Configuration Guide

RTSP Inspection Overview 41-15
Using RealPlayer 41-16
Restrictions and Limitations 41-16
Configuring an RTSP Inspection Policy Map for Additional Inspection Control 41-16

SIP Inspection 41-19
SIP Inspection Overview 41-19
SIP Instant Messaging 41-19
Configuring a SIP Inspection Policy Map for Additional Inspection Control 41-20
Configuring SIP Timeout Values 41-24
Verifying and Monitoring SIP Inspection 41-24

Skinny (SCCP) Inspection 41-25
SCCP Inspection Overview 41-25
Supporting Cisco IP Phones 41-26
Restrictions and Limitations 41-26
Configuring a Skinny (SCCP) Inspection Policy Map for Additional Inspection Control 41-26
Verifying and Monitoring SCCP Inspection 41-28

CHAPTER 42
Configuring Inspection of Database and Directory Protocols 42-1
ILS Inspection 42-1
SQL*Net Inspection 42-2
Sun RPC Inspection 42-3
Sun RPC Inspection Overview 42-3
Managing Sun RPC Services 42-4
Verifying and Monitoring Sun RPC Inspection 42-4

CHAPTER 43
Configuring Inspection for Management Application Protocols 43-1
DCERPC Inspection 43-1
DCERPC Overview 43-1
Configuring a DCERPC Inspection Policy Map for Additional Inspection Control 43-2
GTP Inspection 43-3
GTP Inspection Overview 43-3
Configuring a GTP Inspection Policy Map for Additional Inspection Control 43-4
Verifying and Monitoring GTP Inspection 43-8
RADIUS Accounting Inspection 43-9
RADIUS Accounting Inspection Overview 43-9
Configuring a RADIUS Inspection Overview 43-9
Configuring a RADIUS Inspection Policy Map for Additional Inspection Control 43-10
RSH Inspection 43-11
SNMP Inspection 43-11
SNMP Inspection Overview 43-11
Configuring an SNMP Inspection Policy Map for Additional Inspection Control 43-11
XDMCP Inspection 43-12

Configuring Connection Settings and QoS

Configuring Connection Settings 44-1
  Information About Connection Settings 44-1
    TCP Intercept and Limiting Embryonic Connections 44-2
    Disabling TCP Intercept for Management Packets for Clientless SSL Compatibility 44-2
    Dead Connection Detection (DCD) 44-2
    TCP Sequence Randomization 44-3
    TCP Normalization 44-3
    TCP State Bypass 44-3
  Licensing Requirements for Connection Settings 44-4
  Guidelines and Limitations 44-5
    TCP State Bypass Guidelines and Limitations 44-5
  Default Settings 44-5
  Configuring Connection Settings 44-6
    Task Flow For Configuring Configuration Settings (Except Global Timeouts) 44-6
    Customizing the TCP Normalizer with a TCP Map 44-6
    Configuring Connection Settings 44-10
  Monitoring Connection Settings 44-14
    Monitoring TCP State Bypass 44-14
  Configuration Examples for Connection Settings 44-14
    Configuration Examples for Connection Limits and Timeouts 44-15
    Configuration Examples for TCP State Bypass 44-15
    Configuration Examples for TCP Normalization 44-15
  Feature History for Connection Settings 44-16

Configuring QoS 45-1
  Information About QoS 45-1
    Supported QoS Features 45-2
    What is a Token Bucket? 45-2
    Information About Policing 45-2
    DSCP and DiffServ Preservation 45-2
  Licensing Requirements for QoS 45-3
  Guidelines and Limitations 45-3
Configuring QoS 45-3
  Configuring a Service Rule for Policing 45-3
Monitoring QoS 45-6
  Viewing QoS Police Statistics 45-6
Feature History for QoS 45-6

PART 12 Configuring Advanced Network Protection

CHAPTER 46 Configuring the Botnet Traffic Filter 46-1
  Information About the Botnet Traffic Filter 46-1
    Botnet Traffic Filter Address Types 46-2
    Botnet Traffic Filter Actions for Known Addresses 46-2
    Botnet Traffic Filter Databases 46-2
      Information About the Dynamic Database 46-2
      Information About the Static Database 46-3
      Information About the DNS Reverse Lookup Cache and DNS Host Cache 46-4
    How the Botnet Traffic Filter Works 46-5
  Licensing Requirements for the Botnet Traffic Filter 46-6
  Guidelines and Limitations 46-6
  Default Settings 46-6
  Configuring the Botnet Traffic Filter 46-6
    Task Flow for Configuring the Botnet Traffic Filter 46-7
    Configuring the Dynamic Database 46-7
    Adding Entries to the Static Database 46-9
    Enabling DNS Snooping 46-10
    Enabling Traffic Classification and Actions for the Botnet Traffic Filter 46-12
    Blocking Botnet Traffic Manually 46-15
    Searching the Dynamic Database 46-16
  Monitoring the Botnet Traffic Filter 46-17
    Botnet Traffic Filter Syslog Messaging 46-17
    Botnet Traffic Filter Commands 46-17
  Configuration Examples for the Botnet Traffic Filter 46-19
    Recommended Configuration Example 46-19
    Other Configuration Examples 46-20
  Where to Go Next 46-21
  Feature History for the Botnet Traffic Filter 46-22
CHAPTER 47

Configuring Threat Detection 47-1
  Information About Threat Detection 47-1
  Licensing Requirements for Threat Detection 47-1
  Configuring Basic Threat Detection Statistics 47-2
    Information About Basic Threat Detection Statistics 47-2
    Guidelines and Limitations 47-3
    Default Settings 47-3
    Configuring Basic Threat Detection Statistics 47-4
    Monitoring Basic Threat Detection Statistics 47-5
    Feature History for Basic Threat Detection Statistics 47-6
  Configuring Advanced Threat Detection Statistics 47-6
    Information About Advanced Threat Detection Statistics 47-6
    Guidelines and Limitations 47-7
    Default Settings 47-7
    Configuring Advanced Threat Detection Statistics 47-7
    Monitoring Advanced Threat Detection Statistics 47-9
    Feature History for Advanced Threat Detection Statistics 47-14
  Configuring Scanning Threat Detection 47-15
    Information About Scanning Threat Detection 47-15
    Guidelines and Limitations 47-16
    Default Settings 47-16
    Configuring Scanning Threat Detection 47-17
    Monitoring Shunned Hosts, Attackers, and Targets 47-17
    Feature History for Scanning Threat Detection 47-18
  Configuration Examples for Threat Detection 47-19

CHAPTER 48

Using Protection Tools 48-1
  Preventing IP Spoofing 48-1
  Configuring the Fragment Size 48-2
  Blocking Unwanted Connections 48-2
  Configuring IP Audit for Basic IPS Support 48-3
    Configuring IP Audit 48-3
    IP Audit Signature List 48-4

PART 13

Configuring High Availability

CHAPTER 49

Information About High Availability 49-1
  Introduction to Failover and High Availability 49-1
Failover System Requirements 49-2
  Software Requirements 49-2
  License Requirements 49-2
Failover and Stateful Failover Links 49-2
  Failover Link 49-2
  Stateful Failover Link 49-3
    Failover Interface Speed for Stateful Links 49-4
  Avoiding Interrupted Failover Links 49-4
Active/Active and Active/Standby Failover 49-7
  Determining Which Type of Failover to Use 49-7
Stateless (Regular) and Stateful Failover 49-8
  Stateless (Regular) Failover 49-8
  Stateful Failover 49-8
Intra- and Inter-Chassis Module Placement for the ASA Services Module 49-10
  Intra-Chassis Failover 49-10
  Inter-Chassis Failover 49-10
Transparent Firewall Mode Requirements 49-13
Auto Update Server Support in Failover Configurations 49-14
  Auto Update Process Overview 49-14
  Monitoring the Auto Update Process 49-15
Failover Health Monitoring 49-16
  Unit Health Monitoring 49-17
  Interface Monitoring 49-17
Failover Times 49-18
Failover Messages 49-18
  Failover System Messages 49-18
  Debug Messages 49-19
  SNMP 49-19

Chapter 50
Configuring Active/Standby Failover 50-1
Information About Active/Standby Failover 50-1
  Active/Standby Failover Overview 50-1
  Primary/Secondary Status and Active/Standby Status 50-2
  Device Initialization and Configuration Synchronization 50-2
  Command Replication 50-3
  Failover Triggers 50-4
  Failover Actions 50-4
  Optional Active/Standby Failover Settings 50-6
<table>
<thead>
<tr>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Licensing Requirements for Active/Standby Failover</td>
<td>50-6</td>
</tr>
<tr>
<td>Prerequisites for Active/Standby Failover</td>
<td>50-6</td>
</tr>
<tr>
<td>Guidelines and Limitations</td>
<td>50-6</td>
</tr>
<tr>
<td>Configuring Active/Standby Failover</td>
<td>50-7</td>
</tr>
<tr>
<td>Task Flow for Configuring Active/Standby Failover</td>
<td>50-8</td>
</tr>
<tr>
<td>Configuring the Primary Unit</td>
<td>50-8</td>
</tr>
<tr>
<td>Configuring the Secondary Unit</td>
<td>50-11</td>
</tr>
<tr>
<td>Configuring Optional Active/Standby Failover Settings</td>
<td>50-12</td>
</tr>
<tr>
<td>Enabling HTTP Replication with Stateful Failover</td>
<td>50-13</td>
</tr>
<tr>
<td>Disabling and Enabling Interface Monitoring</td>
<td>50-13</td>
</tr>
<tr>
<td>Configuring Failover Criteria</td>
<td>50-14</td>
</tr>
<tr>
<td>Configuring the Unit and Interface Health Poll Times</td>
<td>50-14</td>
</tr>
<tr>
<td>Configuring Virtual MAC Addresses</td>
<td>50-15</td>
</tr>
<tr>
<td>Controlling Failover</td>
<td>50-16</td>
</tr>
<tr>
<td>Forcing Failover</td>
<td>50-16</td>
</tr>
<tr>
<td>Disabling Failover</td>
<td>50-17</td>
</tr>
<tr>
<td>Restoring a Failed Unit</td>
<td>50-17</td>
</tr>
<tr>
<td>Testing the Failover Functionality</td>
<td>50-17</td>
</tr>
<tr>
<td>Monitoring Active/Standby Failover</td>
<td>50-18</td>
</tr>
<tr>
<td>Feature History for Active/Standby Failover</td>
<td>50-18</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuring Active/Active Failover</td>
<td>51-1</td>
</tr>
<tr>
<td>Information About Active/Active Failover</td>
<td>51-1</td>
</tr>
<tr>
<td>Active/Active Failover Overview</td>
<td>51-1</td>
</tr>
<tr>
<td>Primary/Secondary Status and Active/Standby Status</td>
<td>51-2</td>
</tr>
<tr>
<td>Device Initialization and Configuration Synchronization</td>
<td>51-3</td>
</tr>
<tr>
<td>Command Replication</td>
<td>51-3</td>
</tr>
<tr>
<td>Failover Triggers</td>
<td>51-4</td>
</tr>
<tr>
<td>Failover Actions</td>
<td>51-5</td>
</tr>
<tr>
<td>Optional Active/Active Failover Settings</td>
<td>51-6</td>
</tr>
<tr>
<td>Licensing Requirements for Active/Active Failover</td>
<td>51-6</td>
</tr>
<tr>
<td>Prerequisites for Active/Active Failover</td>
<td>51-6</td>
</tr>
<tr>
<td>Guidelines and Limitations</td>
<td>51-7</td>
</tr>
<tr>
<td>Configuring Active/Active Failover</td>
<td>51-8</td>
</tr>
<tr>
<td>Task Flow for Configuring Active/Active Failover</td>
<td>51-8</td>
</tr>
<tr>
<td>Configuring the Primary Failover Unit</td>
<td>51-8</td>
</tr>
<tr>
<td>Configuring the Secondary Failover Unit</td>
<td>51-11</td>
</tr>
<tr>
<td>Configuring Optional Active/Active Failover Settings</td>
<td>51-12</td>
</tr>
</tbody>
</table>
Configuring Failover Group Preemption... 51-12
Enabling HTTP Replication with Stateful Failover... 51-14
Disabling and Enabling Interface Monitoring... 51-14
Configuring Interface Health Monitoring... 51-15
Configuring Failover Criteria... 51-16
Configuring Virtual MAC Addresses... 51-16
Configuring Support for Asymmetrically Routed Packets... 51-18
Remote Command Execution... 51-21
Changing Command Modes... 51-22
Security Considerations... 51-23
Limitations of Remote Command Execution... 51-23
Controlling Failover... 51-23
Forcing Failover... 51-24
Disabling Failover... 51-24
Restoring a Failed Unit or Failover Group... 51-24
Testing the Failover Functionality... 51-24
Monitoring Active/Active Failover... 51-25
Feature History for Active/Active Failover... 51-25

PART 14
Configuring Logging, SNMP, and Smart Call Home

CHAPTER 52
Configuring Logging 52-1
Information About Logging 52-1
Logging in Multiple Context Mode 52-2
Analyzing Syslog Messages 52-2
Syslog Message Format 52-3
Severity Levels 52-3
Message Classes and Range of Syslog IDs 52-4
Filtering Syslog Messages 52-4
Using Custom Message Lists 52-4
Licensing Requirements for Logging 52-5
Prerequisites for Logging 52-5
Guidelines and Limitations 52-5
Configuring Logging 52-6
Enabling Logging 52-6
Configuring an Output Destination 52-6
Sending Syslog Messages to an External Syslog Server 52-8
Sending Syslog Messages to the Internal Log Buffer 52-9
Contents

Sendings Syslog Messages to an E-mail Address 52-10
Sending Syslog Messages to ASDM 52-11
Sending Syslog Messages to the Console Port 52-11
Sending Syslog Messages to an SNMP Server 52-12
Sending Syslog Messages to a Telnet or SSH Session 52-12
Creating a Custom Event List 52-13
Generating Syslog Messages in EMBLEM Format to a Syslog Server 52-14
Generating Syslog Messages in EMBLEM Format to Other Output Destinations 52-14
Changing the Amount of Internal Flash Memory Available for Logs 52-15
Configuring the Logging Queue 52-15
Sending All Syslog Messages in a Class to a Specified Output Destination 52-16
Enabling Secure Logging 52-16
Including the Device ID in Non-EMBLEM Format Syslog Messages 52-17
Including the Date and Time in Syslog Messages 52-18
Disabling a Syslog Message 52-18
Changing the Severity Level of a Syslog Message 52-18
Limiting the Rate of Syslog Message Generation 52-19

Monitoring the Logs 52-19
Configuration Examples for Logging 52-20
Feature History for Logging 52-20

CHAPTER 53

Configuring NetFlow Secure Event Logging (NSEL) 53-1

Information About NSEL 53-1
Using NSEL and Syslog Messages 53-2
Licensing Requirements for NSEL 53-3
Prerequisites for NSEL 53-3
Guidelines and Limitations 53-4
Configuring NSEL 53-4
Configuring NSEL Collectors 53-5
Configuring Flow-Export Actions Through Modular Policy Framework 53-5
Configuring Template Timeout Intervals 53-7
Changing the Time Interval for Sending Flow-Update Events to a Collector 53-8
Delaying Flow-Create Events 53-9
Disabling and Reenabling NetFlow-related Syslog Messages 53-9
Clearing Runtime Counters 53-10
Monitoring NSEL 53-10
NSEL Monitoring Commands 53-10
Configuration Examples for NSEL 53-12
Configuration Example for SNMP Version 3  54-27
Where to Go Next  54-28
Additional References  54-28
  RFCs for SNMP Version 3  54-28
  MIBs  54-28
  Application Services and Third-Party Tools  54-30
Feature History for SNMP  54-30

Chapter 55
Configuring Anonymous Reporting and Smart Call Home  55-1
  Information About Anonymous Reporting and Smart Call Home  55-1
    Information About Anonymous Reporting  55-2
      What is Sent to Cisco?  55-2
      DNS Requirement  55-3
    Anonymous Reporting and Smart Call Home Prompt  55-3
  Information About Smart Call Home  55-4
  Licensing Requirements for Anonymous Reporting and Smart Call Home  55-4
  Prerequisites for Smart Call Home and Anonymous Reporting  55-5
  Guidelines and Limitations  55-5
  Configuring Anonymous Reporting and Smart Call Home  55-6
    Configuring Anonymous Reporting  55-6
    Configuring Smart Call Home  55-7
      Enabling Smart Call Home  55-7
      Declaring and Authenticating a CA Trust Point  55-8
      Configuring DNS  55-8
      Subscribing to Alert Groups  55-9
      Testing Call Home Communications  55-11
    Optional Configuration Procedures  55-13
  Monitoring Smart Call Home  55-19
  Configuration Example for Smart Call Home  55-19
  Feature History for Anonymous Reporting and Smart Call Home  55-20

Part 15
System Administration

Chapter 56
Managing Software and Configurations  56-1
  Managing the Flash File System  56-1
    Viewing Files in Flash Memory  56-1
    Deleting Files from Flash Memory  56-2
  Downloading Software or Configuration Files to Flash Memory  56-2
Contents

Downloading a File to a Specific Location  56-3
Downloading a File to the Startup or Running Configuration  56-3
Configuring the Application Image and ASDM Image to Boot  56-4
Configuring the File to Boot as the Startup Configuration  56-5
Deleting Files from a USB Drive on the ASA 5500-X Series  56-5
Performing Zero Downtime Upgrades for Failover Pairs  56-6
Upgrading an Active/Standby Failover Configuration  56-6
Upgrading an Active/Active Failover Configuration  56-7
Backing Up Configuration Files or Other Files  56-8
   Backing up the Single Mode Configuration or Multiple Mode System Configuration  56-8
   Backing Up a Context Configuration or Other File in Flash Memory  56-8
   Backing Up a Context Configuration within a Context  56-9
   Copying the Configuration from the Terminal Display  56-9
   Backing Up Additional Files Using the Export and Import Commands  56-9
   Using a Script to Back Up and Restore Files  56-10
      Prerequisites  56-10
      Running the Script  56-10
      Sample Script  56-11
Configuring Auto Update Support  56-16
   Configuring Communication with an Auto Update Server  56-16
   Configuring Client Updates as an Auto Update Server  56-18
   Viewing Auto Update Status  56-19
Downgrading Your Software  56-19
   Information About Activation Key Compatibility  56-20
   Performing the Downgrade  56-20

Chapter 57  57-1
Troubleshooting  57-1
Testing Your Configuration  57-1
   Enabling ICMP Debugging Messages and Syslog Messages  57-2
   Pinging ASASM Interfaces  57-3
   Passing Traffic Through the ASASM  57-5
   Disabling the Test Configuration  57-7
   Determining Packet Routing with Traceroute  57-7
   Tracing Packets with Packet Tracer  57-7
   Handling TCP Packet Loss  57-8
Reloading the ASASM  57-8
Performing Password Recovery  57-8
   Recovering Passwords for the ASASM  57-9
Disabling Password Recovery 57-10
Resetting the Password on the SSM Hardware Module 57-11
Using the ROM Monitor to Load a Software Image 57-11
Erasing the Flash File System 57-12
Other Troubleshooting Tools 57-13
Viewing Debugging Messages 57-13
Capturing Packets 57-14
Viewing the Crash Dump 57-14
Coredump 57-14
Monitoring Per-Process CPU Usage 57-14
Common Problems 57-14

Reference

APPENDIX A

Using the Command-Line Interface A-1
Firewall Mode and Security Context Mode A-1
Command Modes and Prompts A-2
Syntax Formatting A-3
Abbreviating Commands A-3
Command-Line Editing A-3
Command Completion A-4
Command Help A-4
Filtering show Command Output A-4
Command Output Paging A-5
Adding Comments A-5
Text Configuration Files A-5
How Commands Correspond with Lines in the Text File A-6
Command-Specific Configuration Mode Commands A-6
Automatic Text Entries A-7
Line Order A-7
Commands Not Included in the Text Configuration A-7
Passwords A-7
Multiple Security Context Files A-7
Supported Character Sets A-8

APPENDIX B

Addresses, Protocols, and Ports B-1
IPv4 Addresses and Subnet Masks B-1
About This Guide

This preface introduces Cisco ASA Services Module CLI Configuration Guide and includes the following sections:

- Document Objectives, page xlv
- Audience, page xlv
- Related Documentation, page xlv
- Conventions, page xlvi
- Obtaining Documentation and Submitting a Service Request, page xlvii

Document Objectives

The purpose of this guide is to help you configure the ASASM using the command-line interface. This guide does not cover every feature, but describes only the most common configuration scenarios.

You can also configure and monitor the ASASM by using ASDM, a web-based GUI application. ASDM includes configuration wizards to guide you through some common configuration scenarios, and online help for less common scenarios.

This guide applies to the ASA Services Module.

Audience

This guide is for network managers who perform any of the following tasks:

- Manage network security
- Install and configure firewalls/ASASMs

Related Documentation

Conventions

This document uses the following conventions:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Indication</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>bold</strong> font</td>
<td>Commands and keywords and user-entered text appear in <strong>bold</strong> font.</td>
</tr>
<tr>
<td><em>italic</em> font</td>
<td>Document titles, new or emphasized terms, and arguments for which you supply values are in <em>italic</em> font.</td>
</tr>
<tr>
<td>[  ]</td>
<td>Elements in square brackets are optional.</td>
</tr>
<tr>
<td>{ x</td>
<td>y</td>
</tr>
<tr>
<td>[ x</td>
<td>y</td>
</tr>
<tr>
<td>string</td>
<td>A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.</td>
</tr>
<tr>
<td><strong>courier</strong> font</td>
<td>Terminal sessions and information the system displays appear in <strong>courier</strong> font.</td>
</tr>
<tr>
<td>&lt; &gt;</td>
<td>Nonprinting characters such as passwords are in angle brackets.</td>
</tr>
<tr>
<td>[  ]</td>
<td>Default responses to system prompts are in square brackets.</td>
</tr>
<tr>
<td>!, #</td>
<td>An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.</td>
</tr>
</tbody>
</table>

**Note**

Means reader take note.

**Tip**

Means the following information will help you solve a problem.

**Caution**

Means reader be careful. In this situation, you might perform an action that could result in equipment damage or loss of data.

**Timesaver**

Means the described action saves time. You can save time by performing the action described in the paragraph.

**Warning**

Means reader be warned. In this situation, you might perform an action that could result in bodily injury.
Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, see the monthly What's New in Cisco Product Documentation, which also lists all new and revised Cisco technical documentation, at:


Subscribe to the What's New in Cisco Product Documentation as an RSS feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service. Cisco currently supports RSS Version 2.0.
PART 1

Getting Started with the ASA
Introduction to the Cisco ASA Services Module

The ASASM provides advanced Stateful Firewall functionality. The ASASM includes many advanced features, such as multiple security contexts (similar to virtualized firewalls), transparent (Layer 2) firewall or routed (Layer 3) firewall operation, advanced inspection engines, and many more features.

This chapter includes the following sections:

- Hardware and Software Compatibility, page 1-1
- New Features, page 1-1
- How the ASA Services Module Works with the Switch, page 1-5
- Firewall Functional Overview, page 1-7
- Security Context Overview, page 1-11

Hardware and Software Compatibility

For a complete list of supported hardware and software, see the Cisco ASA Compatibility:

New Features

This section includes the following topics:

- New Features in Version 8.5(1.7), page 1-2
- New Features in Version 8.5(1.6), page 1-2
- New Features in Version 8.5(1), page 1-3

Note
New, changed, and deprecated syslog messages are listed in syslog messages guide.

Note
Version 8.4(4) was removed from Cisco.com due to build issues; please upgrade to Version 8.4(4.1) or later.
Chapter 1      Introduction to the Cisco ASA Services Module

New Features in Version 8.5(1.7)

Released: March 5, 2012

Table 1-2 lists the new features for ASA interim Version 8.5(1.7).

Note

We recommend that you upgrade to a Cisco.com-posted ASA interim release only if you have a specific problem that it resolves. If you decide to run an interim release in a production environment, keep in mind that only targeted testing is performed on interim releases. Interim releases are fully supported by Cisco TAC and will usually remain on the download site only until the next maintenance release is available. If you choose to run an interim release, we strongly encourage you to upgrade to a fully-tested maintenance or feature release when it becomes available.

We will document interim release features at the time of the next maintenance or feature release. For a list of resolved caveats for each ASA interim release, see the interim release notes available on the Cisco.com software download site.

<table>
<thead>
<tr>
<th>Feature Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Hardware Features</strong></td>
<td></td>
</tr>
<tr>
<td>Support for the Catalyst 6500 Supervisor 2T</td>
<td>The ASASM now interoperates with the Catalyst 6500 Supervisor 2T. For hardware and software compatibility, see: <a href="http://www.cisco.com/en/US/docs/security/asa/compatibility/asamatrix.html">http://www.cisco.com/en/US/docs/security/asa/compatibility/asamatrix.html</a>.</td>
</tr>
<tr>
<td><strong>Note</strong></td>
<td>You may have to upgrade the FPD image on the ASASM. See the Upgrading procedure the in the release notes.</td>
</tr>
<tr>
<td><strong>Failover Features</strong></td>
<td></td>
</tr>
<tr>
<td>Configure the connection replication rate during a bulk sync</td>
<td>You can now configure the rate at which the ASASM replicates connections to the standby unit when using stateful failover. By default, connections are replicated to the standby unit during a 15 second period. However, when a bulk sync occurs (for example, when you first enable failover), 15 seconds may not be long enough to sync large numbers of connections due to a limit on the maximum connections per second. For example, the maximum connections on the ASASM is 8 million; replicating 8 million connections in 15 seconds means creating 533K connections per second. However, the maximum connections allowed per second is 300K. You can now specify the rate of replication to be less than or equal to the maximum connections per second, and the sync period will be adjusted until all the connections are synced.</td>
</tr>
<tr>
<td></td>
<td>We introduced the following command: <code>failover replication rate rate</code>.</td>
</tr>
</tbody>
</table>

New Features in Version 8.5(1.6)

Released: January 27, 2012
Table 1-2 lists the new features for ASA interim Version 8.5(1.6)/ASDM Version 6.5(1).

We recommend that you upgrade to a Cisco.com-posted ASA interim release only if you have a specific problem that it resolves. If you decide to run an interim release in a production environment, keep in mind that only targeted testing is performed on interim releases. Interim releases are fully supported by Cisco TAC and will usually remain on the download site only until the next maintenance release is available. If you choose to run an interim release, we strongly encourage you to upgrade to a fully-tested maintenance or feature release when it becomes available.

We will document interim release features at the time of the next maintenance or feature release. For a list of resolved caveats for each ASA interim release, see the interim release notes available on the Cisco.com software download site.

Table 1-2  New Features for ASA Interim Version 8.5(1.6)

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multiple Context Features</td>
<td>In multiple context mode, the ASASM now converts the automatic MAC address generation configuration to use a default prefix. The ASASM auto-generates the prefix based on the last two bytes of the backplane MAC address. This conversion happens automatically when you reload, or if you reenable MAC address generation. The prefix method of generation provides many benefits, including a better guarantee of unique MAC addresses on a segment. You can view the auto-generated prefix by entering the show running-config mac-address command. If you want to change the prefix, you can reconfigure the feature with a custom prefix. The legacy method of MAC address generation is no longer available.</td>
</tr>
<tr>
<td>Note</td>
<td>To maintain hitless upgrade for failover pairs, the ASASM does not convert the MAC address method in an existing configuration upon a reload if failover is enabled. However, we strongly recommend that you manually change to the prefix method of generation when using failover. Without the prefix method, ASASMs installed in different slot numbers experience a MAC address change upon failover, and can experience traffic interruption. After upgrading, to use the prefix method of MAC address generation, reenable MAC address generation to use the default prefix.</td>
</tr>
<tr>
<td></td>
<td>We modified the following command: <strong>mac-address auto</strong>.</td>
</tr>
</tbody>
</table>

New Features in Version 8.5(1)

Released: July 8, 2011

Table 1-2 lists the new features for ASA Version 8.5(1). This ASA software version is only supported on the ASASM.

**Note**  Version 8.5(1) includes all features in 8.4(1), plus the features listed in this table. The following features, however, are not supported in No Payload Encryption software, and this release is only available as a No Payload Encryption release:
- VPN
- Unified Communications

Features added in 8.4(2) are not included in 8.5(1) unless they are explicitly listed in this table.

### Table 1-3  New Features for ASA Version 8.5(1)

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Hardware Features</strong></td>
<td></td>
</tr>
<tr>
<td>Support for the ASA Services Module</td>
<td>We introduced support for the ASASM for the Cisco Catalyst 6500 E switch.</td>
</tr>
<tr>
<td><strong>Firewall Features</strong></td>
<td></td>
</tr>
<tr>
<td>Mixed firewall mode support in multiple context mode</td>
<td>You can set the firewall mode independently for each security context in multiple context mode, so some can run in transparent mode while others run in routed mode. We modified the following command: firewall transparent.</td>
</tr>
<tr>
<td><strong>Interface Features</strong></td>
<td></td>
</tr>
<tr>
<td>Automatic MAC address generation is now enabled by default in multiple context mode</td>
<td>Automatic generation of MAC addresses is now enabled by default in multiple context mode. We modified the following command: mac address auto.</td>
</tr>
<tr>
<td><strong>NAT Features</strong></td>
<td></td>
</tr>
<tr>
<td>Identity NAT configurable proxy ARP and route lookup</td>
<td>In earlier releases for identity NAT, proxy ARP was disabled, and a route lookup was always used to determine the egress interface. You could not configure these settings. In 8.4(2) and later, the default behavior for identity NAT was changed to match the behavior of other static NAT configurations: proxy ARP is enabled, and the NAT configuration determines the egress interface (if specified) by default. You can leave these settings as is, or you can enable or disable them discretely. Note that you can now also disable proxy ARP for regular static NAT. For pre-8.3 configurations, the migration of NAT exempt rules (the nat 0 access-list command) to 8.4(2) and later now includes the following keywords to disable proxy ARP and to use a route lookup: no-proxy-arp and route-lookup. The unidirectional keyword that was used for migrating to 8.3(2) and 8.4(1) is no longer used for migration. When upgrading to 8.4(2) from 8.3(1), 8.3(2), and 8.4(1), all identity NAT configurations will now include the no-proxy-arp and route-lookup keywords, to maintain existing functionality. The unidirectional keyword is removed. We modified the following commands: nat static [no-proxy-arp] [route-lookup] (object network) and nat source static [no-proxy-arp] [route-lookup] (global). Also available in Version 8.4(2).</td>
</tr>
</tbody>
</table>

Also available in Version 8.4(2).
You can install the ASASM in the Catalyst 6500 series switches with Cisco IOS software on both the switch supervisor and the integrated MSFC.

**Note**  The Catalyst Operating System (OS) is not supported.

The ASASM runs its own operating system
The switch includes a switching processor (Applythe supervisor) and a router (the MSFC). Although you need the MSFC as part of your system, you do not have to use it. If you choose to do so, you can assign one or more VLAN interfaces to the MSFC. You can alternatively use external routers instead of the MSFC.

In single context mode, you can place the router in front of the firewall or behind the firewall (see Figure 1).

The location of the router depends entirely on the VLANs that you assign to it. For example, the router is behind the firewall in the example shown on the left side of Figure 1 because you assigned VLAN 201 to the inside interface of the ASASM. The router is in front of the firewall in the example shown on the right side of Figure 1 because you assigned VLAN 200 to the outside interface of the ASASM.

In the left-hand example, the MSFC or router routes between VLANs 201, 301, 302, and 303, and no inside traffic goes through the ASASM unless it is destined for the Internet. In the right-hand example, the ASASM processes and protects all traffic between the inside VLANs 201, 202, and 203.

**Figure 1**  MSFC/Router Placement
For multiple context mode, if you place the router behind the ASASM, you should only connect it to a single context. If you connect the router to multiple contexts, the router will route between the contexts, which might not be your intention. The typical scenario for multiple contexts is to use a router in front of all the contexts to route between the Internet and the switched networks (see Figure 2).

Figure 2  MSFC/Router Placement with Multiple Contexts

Firewall Functional Overview

Firewalls protect inside networks from unauthorized access by users on an outside network. A firewall can also protect inside networks from each other, for example, by keeping a human resources network separate from a user network. If you have network resources that need to be available to an outside user, such as a web or FTP server, you can place these resources on a separate network behind the firewall, called a demilitarized zone (DMZ). The firewall allows limited access to the DMZ, but because the DMZ only includes the public servers, an attack there only affects the servers and does not affect the other inside networks. You can also control when inside users access outside networks (for example, access to the Internet), by allowing only certain addresses out, by requiring authentication or authorization, or by coordinating with an external URL filtering server.

When discussing networks connected to a firewall, the outside network is in front of the firewall, the inside network is protected and behind the firewall, and a DMZ, while behind the firewall, allows limited access to outside users. Because the ASASM lets you configure many interfaces with varied security policies, including many inside interfaces, many DMZs, and even many outside interfaces if desired, these terms are used in a general sense only.
Security Policy Overview

A security policy determines which traffic is allowed to pass through the firewall to access another network. By default, the ASASM allows traffic to flow freely from an inside network (higher security level) to an outside network (lower security level). You can apply actions to traffic to customize the security policy. This section includes the following topics:

- Permitting or Denying Traffic with Access Lists, page 1-8
- Applying NAT, page 1-8
- Protecting from IP Fragments, page 1-8
- Using AAA for Through Traffic, page 1-9
- Applying HTTP, HTTPS, or FTP Filtering, page 1-9
- Applying Application Inspection, page 1-9
- Applying QoS Policies, page 1-9
- Applying Connection Limits and TCP Normalization, page 1-9
- Enabling Threat Detection, page 1-9
- Enabling the Botnet Traffic Filter, page 1-10

Permitting or Denying Traffic with Access Lists

You can apply an access list to limit traffic from inside to outside, or allow traffic from outside to inside. For transparent firewall mode, you can also apply an EtherType access list to allow non-IP traffic.

Applying NAT

Some of the benefits of NAT include the following:

- You can use private addresses on your inside networks. Private addresses are not routable on the Internet.
- NAT hides the local addresses from other networks, so attackers cannot learn the real address of a host.
- NAT can resolve IP routing problems by supporting overlapping IP addresses.

Protecting from IP Fragments

The ASASM provides IP fragment protection. This feature performs full reassembly of all ICMP error messages and virtual reassembly of the remaining IP fragments that are routed through the ASASM. Fragments that fail the security check are dropped and logged. Virtual reassembly cannot be disabled.
Using AAA for Through Traffic

You can require authentication and/or authorization for certain types of traffic, for example, for HTTP. The ASASM also sends accounting information to a RADIUS or TACACS+ server.

Applying HTTP, HTTPS, or FTP Filtering

Although you can use access lists to prevent outbound access to specific websites or FTP servers, configuring and managing web usage this way is not practical because of the size and dynamic nature of the Internet. We recommend that you use the ASASM in conjunction with a separate server running one of the following Internet filtering products:

- Websense Enterprise
- Secure Computing SmartFilter

Applying Application Inspection

Inspection engines are required for services that embed IP addressing information in the user data packet or that open secondary channels on dynamically assigned ports. These protocols require the ASASM to perform a deep packet inspection.

Applying QoS Policies

Some network traffic, such as voice and streaming video, cannot tolerate long latency times. QoS is a network feature that lets you give priority to these types of traffic. QoS refers to the capability of a network to provide better service to selected network traffic.

Applying Connection Limits and TCP Normalization

You can limit TCP and UDP connections and embryonic connections. Limiting the number of connections and embryonic connections protects you from a DoS attack. The ASASM uses the embryonic limit to trigger TCP Intercept, which protects inside systems from a DoS attack perpetrated by flooding an interface with TCP SYN packets. An embryonic connection is a connection request that has not finished the necessary handshake between source and destination.

TCP normalization is a feature consisting of advanced TCP connection settings designed to drop packets that do not appear normal.

Enabling Threat Detection

You can configure scanning threat detection and basic threat detection, and also how to use statistics to analyze threats.

Basic threat detection detects activity that might be related to an attack, such as a DoS attack, and automatically sends a system log message.

A typical scanning attack consists of a host that tests the accessibility of every IP address in a subnet (by scanning through many hosts in the subnet or sweeping through many ports in a host or subnet). The scanning threat detection feature determines when a host is performing a scan. Unlike IPS scan detection that is based on traffic signatures, the ASASM scanning threat detection feature maintains an extensive database that contains host statistics that can be analyzed for scanning activity.
The host database tracks suspicious activity such as connections with no return activity, access of closed service ports, vulnerable TCP behaviors such as non-random IPID, and many more behaviors.

You can configure the ASASM to send system log messages about an attacker or you can automatically shun the host.

### Enabling the Botnet Traffic Filter

Malware is malicious software that is installed on an unknowing host. Malware that attempts network activity such as sending private data (passwords, credit card numbers, key strokes, or proprietary data) can be detected by the Botnet Traffic Filter when the malware starts a connection to a known bad IP address. The Botnet Traffic Filter checks incoming and outgoing connections against a dynamic database of known bad domain names and IP addresses (the blacklist), and then logs any suspicious activity. When you see syslog messages about the malware activity, you can take steps to isolate and disinfect the host.

### Firewall Mode Overview

The ASASM runs in two different firewall modes:

- **Routed**
- **Transparent**

In routed mode, the ASASM is considered to be a router hop in the network.

In transparent mode, the ASASM acts like a “bump in the wire,” or a “stealth firewall,” and is not considered a router hop. The ASASM connects to the same network on its inside and outside interfaces.

You might use a transparent firewall to simplify your network configuration. Transparent mode is also useful if you want the firewall to be invisible to attackers. You can also use a transparent firewall for traffic that would otherwise be blocked in routed mode. For example, a transparent firewall can allow multicast streams using an EtherType access list.

### Stateful Inspection Overview

All traffic that goes through the ASASM is inspected using the Adaptive Security Algorithm and either allowed through or dropped. A simple packet filter can check for the correct source address, destination address, and ports, but it does not check that the packet sequence or flags are correct. A filter also checks every packet against the filter, which can be a slow process.

---

**Note**

The TCP state bypass feature allows you to customize the packet flow. See the “TCP State Bypass” section on page 44-3.

---

A stateful firewall like the ASASM, however, takes into consideration the state of a packet:

- **Is this a new connection?**

  If it is a new connection, the ASASM has to check the packet against access lists and perform other tasks to determine if the packet is allowed or denied. To perform this check, the first packet of the session goes through the “session management path,” and depending on the type of traffic, it might also pass through the “control plane path.”

  The session management path is responsible for the following tasks:

  - Performing the access list checks
Security Context Overview

You can partition a single ASASM into multiple virtual devices, known as security contexts. Each context is an independent device, with its own security policy, interfaces, and administrators. Multiple contexts are similar to having multiple standalone devices. Many features are supported in multiple context mode, including routing tables, firewall features, IPS, and management. Some features are not supported, including VPN and dynamic routing protocols.

In multiple context mode, the ASASM includes a configuration for each context that identifies the security policy, interfaces, and almost all the options you can configure on a standalone device. The system administrator adds and manages contexts by configuring them in the system configuration, which, like a single mode configuration, is the startup configuration. The system configuration identifies basic settings for the ASASM. The system configuration does not include any network interfaces or network settings for itself; rather, when the system needs to access network resources (such as downloading the contexts from the server), it uses one of the contexts that is designated as the admin context.

The admin context is just like any other context, except that when a user logs into the admin context, then that user has system administrator rights and can access the system and all other contexts.
CHAPTER 2

Configuring the Switch for Use with the ASA Services Module

This chapter describes how to configure the Catalyst 6500 series switch for use with the ASASM. Before completing the procedures in this chapter, configure the basic properties of your switch, including assigning VLANs to switch ports, according to the documentation that came with your switch.

This chapter includes the following sections:

- Information About the Switch, page 2-1
- Guidelines and Limitations, page 2-2
- Verifying the Module Installation, page 2-3
- Assigning VLANs to the ASA Services Module, page 2-4
- Using the MSFC as a Directly Connected Router, page 2-5
- Configuring the Switch for ASA Failover, page 2-9
- Resetting the ASA Services Module, page 2-11
- Monitoring the ASA Services Module, page 2-11
- Feature History for the Switch for Use with the ASA Services Module, page 2-13

Information About the Switch

You can install the ASASM in the Catalyst 6500 series switches. The switch includes a switch (the supervisor engine) as well as a router (the MSFC). The connection between the ASASM and the switch is a single 20-GB interface.

The switch supports Cisco IOS software on both the switch supervisor engine and the integrated MSFC router.

Note

The Catalyst operating system software is not supported.

The ASASM runs its own operating system.

Note

Because the ASASM runs its own operating system, upgrading the Cisco IOS software does not affect the operation of the ASASM.
To view a matrix of hardware and software compatibility for the ASASM and Cisco IOS versions, see the Cisco ASA 5500 Series Hardware and Software Compatibility:


Some ASASM features interact with Cisco IOS features. The following features involve Cisco IOS software:

- Virtual Switching System (VSS)—No ASASM configuration is required.
- Autostate—The supervisor informs the ASASM when the last interface on a given VLAN has gone down, which assists in determining whether or not a failover switch is required.
- Clearing entries in the supervisor MAC address table on a failover switch—No ASASM configuration is required.
- Version compatibility—The ASASM will be automatically powered down if the supervisor/ASASM version compatibility matrix check fails.

### Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

**VLAN Guidelines and Limitations**

- Use VLAN IDs 2 to 1000 and from 1025 to 4094.
- Routed ports and WAN ports consume internal VLANs, so it is possible that VLANs in the 1020-1100 range might already be in use.
- You cannot use reserved VLANs.
- You cannot use VLAN 1.
- If you are using ASASM failover within the same switch chassis, do not assign the VLAN(s) that you are reserving for failover and stateful communications to a switch port. However, if you are using failover between chassis, you must include the VLANs in the trunk port between the chassis.
- If you do not add the VLANS to the switch before you assign them to the ASASM, the VLANS are stored in the supervisor engine database and are sent to the ASASM as soon as they are added to the switch.
- You can configure a VLAN in the ASASM configuration before it has been assigned on the switch. Note that when the switch sends the VLAN to the ASASM, the VLAN defaults to be administratively up on the ASASM, regardless of whether the you shut them down in the ASASM configuration. You need to shut them down again in this case.

**SPAN Reflector Guidelines**

In Cisco IOS software Version 12.2SXJ1 and earlier, for each ASASM in a switch, the SPAN reflector feature is enabled. This feature allows multicast traffic (and other traffic that requires a central rewrite engine) to be switched when coming from the ASASM. The SPAN reflector feature uses one SPAN session. To disable this feature, enter the following command:

```
Router(config)# no monitor session servicemodule
```
Verifying the Module Installation

To verify that the switch acknowledges the ASASM and has brought it online, enter the following command.

### Detailed Steps

**Command**

```
show module [switch (1 | 2)] [mod-num | all]
```

**Purpose**

Displays module information. For a switch in a VSS, enter the `switch` keyword.

**Example:**

```
Router# show module 1
```

**Ensure that the Status column shows “Ok” for the ASASM.**

### Examples

The following is sample output from the `show module` command:

```
Router# show module

<table>
<thead>
<tr>
<th>Mod</th>
<th>Ports</th>
<th>Card Type</th>
<th>Model</th>
<th>Serial No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>3</td>
<td>ASA Service Module</td>
<td>WS-SVC-ASA-SM1</td>
<td>SAD143502E8</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>ASA Service Module</td>
<td>WS-SVC-ASA-SM1</td>
<td>SAD135101Z9</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>Supervisor Engine 720 10GE (Active)</td>
<td>VS-S720-10G</td>
<td>SAL12426KB1</td>
</tr>
<tr>
<td>6</td>
<td>16</td>
<td>CEF720 16 port 10GE</td>
<td>WS-X6716-10GE</td>
<td>SAL1442WZD1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Mod</th>
<th>Sub-Module</th>
<th>Model</th>
<th>Serial</th>
<th>Hw</th>
<th>Fw</th>
<th>Sw</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>ASA Application Processor</td>
<td>SVC-APP-PROC-1</td>
<td>SAD1436015D</td>
<td>0.202</td>
<td>Other</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>ASA Application Processor</td>
<td>SVC-APP-INT-1</td>
<td>SAD141002AK</td>
<td>0.106</td>
<td>PwrDown</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Policy Feature Card 3</td>
<td>VS-F6K-PFC3C</td>
<td>SAL12437BM2</td>
<td>1.0</td>
<td>Ok</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>MSFC3 Daughterboard</td>
<td>VS-F6K-MSFC3</td>
<td>SAL12426DE3</td>
<td>1.0</td>
<td>Ok</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Distributed Forwarding Card</td>
<td>WS-F6700-DPC3C</td>
<td>SAL1443XRDC</td>
<td>1.4</td>
<td>Ok</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Base PID:**

```
<table>
<thead>
<tr>
<th>Mod</th>
<th>Model</th>
<th>Serial No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>WS-SVC-APP-HW-1</td>
<td>SAD143502E8</td>
</tr>
<tr>
<td>4</td>
<td>TRIFECTA</td>
<td>SAD135101Z9</td>
</tr>
</tbody>
</table>
```

**Mod Online Diag Status**

```
<table>
<thead>
<tr>
<th>Mod</th>
<th>Online Diag Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Pass</td>
</tr>
<tr>
<td>4</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>6</td>
<td>Pass</td>
</tr>
</tbody>
</table>
```

Command Purpose

- `show module`: Displays module information. For a switch in a VSS, enter the `switch` keyword.
- `switch (1 | 2)`: Selects the switch in a VSS.
- `mod-num | all`: Specifies the module number or displays all modules.
Assigning VLANs to the ASA Services Module

This section describes how to assign VLANs to the ASASM. The ASASM does not include any external physical interfaces. Instead, it uses VLAN interfaces. Assigning VLANs to the ASASM is similar to assigning a VLAN to a switch port; the ASASM includes an internal interface to the Switch Fabric Module (if present) or the shared bus.

Prerequisites

See the switch documentation for information about adding VLANs to the switch and assigning them to switch ports.

Guidelines

- You can assign up to 16 firewall VLAN groups to each ASASM. (You can create more than 16 VLAN groups in Cisco IOS software, but only 16 can be assigned per ASASM.) For example, you can assign all the VLANs to one group; or you can create an inside group and an outside group; or you can create a group for each customer.
- There is no limit on the number of VLANs per group, but the ASASM can only use VLANs up to the ASASM system limit (see the ASASM licensing documentation for more information).
- You cannot assign the same VLAN to multiple firewall groups.
- You can assign a single firewall group to multiple ASASMs. VLANs that you want to assign to multiple ASASMs, for example, can reside in a separate group from VLANs that are unique to each ASASM.
- See the “VLAN Guidelines and Limitations” section on page 2-2.
Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** | firewall vlan-group firewall_group vlan_range | Assigns VLANs to a firewall group. The firewall_group argument is an integer. The vlan_range argument can be one or more VLANs (2 to 1000 and from 1025 to 4094) identified in one of the following ways: 
- A single number (n)
- A range (n-x)
Separate numbers or ranges by commas, as shown in the following example: 5,7-10,13,45-100 |
| Example: | Router(config)# firewall vlan-group 50 55-57 | |

| **Step 2** | firewall [switch (1 | 2)] module slot vlan-group firewall_group | Assigns the firewall groups to the ASASM. For a switch in a VSS, enter the switch argument. To view the slots where the ASASM is installed, enter the show module command. The firewall_group argument is one or more group numbers, which can be one of the following:
- A single number (n)
- A range (n-x)
Separate numbers or ranges by commas, as shown in the following example: 5,7-10 |
| Example: | Router(config)# firewall module 5 vlan-group 50,52 | |

Examples

The following example shows how to create three firewall VLAN groups: one for each ASASM, and one that includes VLANs assigned to both ASASMs:

Router(config)# firewall vlan-group 50 55-57
Router(config)# firewall vlan-group 51 70-85
Router(config)# firewall vlan-group 52 100
Router(config)# firewall module 5 vlan-group 50,52
Router(config)# firewall module 8 vlan-group 51,52

Using the MSFC as a Directly Connected Router

If you want to use the MSFC as a directly connected router (for example, as the default gateway connected to the ASASM outside interface), then add an ASASM VLAN interface to the MSFC as a switched virtual interface (SVI).

This section includes the following topics:
- Information About SVIs, page 2-6
- Configuring SVIs, page 2-8
Information About SVIs

For security reasons, by default, you can configure one SVI between the MSFC and the ASASM; you can enable multiple SVIs, but be sure you do not misconfigure your network.

For example, with multiple SVIs, you could accidentally allow traffic to pass around the ASASM by assigning both the inside and outside VLANs to the MSFC. (See Figure 2-1.)

**Figure 2-1 Multiple SVI Misconfiguration**
You might need to bypass the ASASM in some network scenarios. Figure 2-2 shows an IPX host on the same Ethernet segment as IP hosts. Because the ASASM in routed firewall mode only handles IP traffic and drops other protocol traffic like IPX (transparent firewall mode can optionally allow non-IP traffic), you might want to bypass the ASASM for IPX traffic. Make sure that you configure the MSFC with an access list that allows only IPX traffic to pass on VLAN 201.

*Figure 2-2 Multiple SVIs for IPX*
For transparent firewalls in multiple context mode, you need to use multiple SVIs because each context requires a unique VLAN on its outside interface (see Figure 2-3). You might also choose to use multiple SVIs in routed mode so that you do not have to share a single VLAN for the outside interface.

**Figure 2-3 Multiple SVIs in Multiple Context Mode**

Configuring SVIs

To add an SVI to the MSFC, perform the following steps.

**Detailed Steps**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Allows you to add more than one SVI to the ASASM.</td>
</tr>
<tr>
<td>(Optional)</td>
<td></td>
</tr>
<tr>
<td><code>firewall multiple-vlan-interfaces</code></td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>Router(config)# firewall multiple-vlan-interfaces</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Adds a VLAN interface to the MSFC.</td>
</tr>
<tr>
<td><code>interface vlan vlan_number</code></td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>Router(config)# interface vlan 55</code></td>
<td></td>
</tr>
</tbody>
</table>
Step 3

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip address address mask</td>
<td>Sets the IP address for this interface on the MSFC.</td>
</tr>
</tbody>
</table>

Example:

```
Router(config-if)# ip address 10.1.1.1 255.255.255.0
```

Step 4

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>no shutdown</td>
<td>Enables the interface.</td>
</tr>
</tbody>
</table>

Example:

```
Router(config-if)# no shutdown
```

Examples

The following example shows a typical configuration with multiple SVIs:

```
Router(config)# firewall vlan-group 50 55-57
Router(config)# firewall vlan-group 51 70-85
Router(config)# firewall module 8 vlan-group 50-51
Router(config)# firewall multiple-vlan-interfaces
Router(config)# interface vlan 55
Router(config-if)# ip address 10.1.1.1 255.255.255.0
Router(config-if)# no shutdown
Router(config-if)# interface vlan 56
Router(config-if)# ip address 10.1.2.1 255.255.255.0
Router(config-if)# no shutdown
Router(config-if)# end
```

Configuring the Switch for ASA Failover

This section includes the following topics:

- Assigning VLANs to the Secondary ASA Services Module, page 2-10
- Adding a Trunk Between a Primary Switch and Secondary Switch, page 2-10
- Ensuring Compatibility with Transparent Firewall Mode, page 2-10
- Enabling Autostate Messaging for Rapid Link Failure Detection, page 2-10
Assigning VLANs to the Secondary ASA Services Module

Because both units require the same access to the inside and outside networks, you must assign the same VLANs to both ASASMs on the switch(es). See the “Assigning VLANs to the Secondary ASA Services Module” section on page 2-10.

Adding a Trunk Between a Primary Switch and Secondary Switch

If you are using inter-switch failover, then you should configure an 802.1Q VLAN trunk between the two switches to carry the failover and state links. The trunk should have QoS enabled so that failover VLAN packets, which have a CoS value of 5 (higher priority), are treated with higher priority in these ports.

To configure the EtherChannel and trunk, see the documentation for your switch.

Ensuring Compatibility with Transparent Firewall Mode

To avoid loops when you use failover in transparent mode, use switch software that supports BPDU forwarding. Do not enable LoopGuard globally on the switch if the ASASM is in transparent mode. LoopGuard is automatically applied to the internal EtherChannel between the switch and the ASASM, so after a failover and a failback, LoopGuard causes the secondary unit to be disconnected because the EtherChannel goes into the err-disable state.

Enabling Autostate Messaging for Rapid Link Failure Detection

The supervisor engine can send autostate messages to the ASASM about the status of physical interfaces associated with ASASM VLANs. For example, when all physical interfaces associated with a VLAN go down, the autostate message tells the ASASM that the VLAN is down. This information lets the ASASM declare the VLAN as down, bypassing the interface monitoring tests normally required for determining which side suffered a link failure. Autostate messaging provides a dramatic improvement in the time the ASASM takes to detect a link failure (a few milliseconds as compared to up to 45 seconds without autostate support).

The switch supervisor sends an autostate message to the ASASM when:

- The last interface belonging to a VLAN goes down.
- The first interface belonging to a VLAN comes up.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>firewall autostate</code></td>
<td>Enables autostate messaging in Cisco IOS software. Autostate messaging is disabled by default.</td>
</tr>
</tbody>
</table>

Example:

```bash
Router(config)# firewall autostate
```
Chapter 2  Configuring the Switch for Use with the ASA Services Module

Reseting the ASA Services Module

This section describes how to reset the ASASM. You might need to reset the ASASM if you cannot reach it through the CLI or an external Telnet session. The reset process might take several minutes.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>hw-module [switch {1</td>
<td>2}] module slot reset</td>
</tr>
</tbody>
</table>

Note: To reset the ASASM when you are already logged in to it, enter either the reload or reboot command.

Examples

The following is sample output from the hw-module module reset command:

Router# hw-module module 9 reset
Proceed with reload of module? [confirm] y
% reset issued for module 9

Router# 00:26:55:%SNMP-5-MODULETRAP:Module 9 [Down] Trap
00:26:55:SP:The PC in slot 8 is shutting down. Please wait ...

Monitoring the ASA Services Module

To monitor the ASASM, enter one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show firewall module [mod-num] state</td>
<td>Verifies the state of the ASASM.</td>
</tr>
<tr>
<td>show firewall module [mod-num] traffic</td>
<td>Verifies that traffic is flowing through the ASASM.</td>
</tr>
<tr>
<td>show firewall module [mod-num] version</td>
<td>Shows the software version of the ASASM.</td>
</tr>
<tr>
<td>show firewall multiple-vlan-interfaces</td>
<td>Indicates the status of multiple VLAN interfaces (enabled or disabled).</td>
</tr>
<tr>
<td>show firewall vlan-group</td>
<td>Displays all configured VLAN groups.</td>
</tr>
<tr>
<td>show interface vlan</td>
<td>Displays the status and information about the configured VLAN interface.</td>
</tr>
</tbody>
</table>

Examples

The following is sample output from the show firewall module [mod-num] state command:
Router> `show firewall module 11 state`
Firewall module 11:
Switchport: Enabled
Administrative Mode: trunk
Operational Mode: trunk
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: dot1q
Negotiation of Trunking: Off
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Pruning VLANs Enabled: 2-1001
Vlans allowed on trunk:
Vlans allowed and active in management domain:
Vlans in spanning tree forwarding state and not pruned:

The following is sample output from the `show firewall module [mod-num] traffic` command:

Router> `show firewall module 11 traffic`
Firewall module 11:

Specified interface is up, line protocol is up (connected)
Hardware is EtherChannel, address is 0014.1cd5.bef6 (bia 0014.1cd5.bef6)
MTU 1500 bytes, BW 6000000 Kbit, DLY 10 usec,
reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA, loopback not set
Full-duplex, 1000Mb/s, media type is unknown
input flow-control is on, output flow-control is on
Members in this channel: Gi11/1 Gi11/2 Gi11/3 Gi11/4 Gi11/5 Gi11/6
Last input never, output never, output hang never
Last clearing of "show interface" counters never
Input queue: 0/2000/0/0 (size/max/drops/flushes); Total output drops: 0
Queuing strategy: fifo
Output queue: 0/40 (size/max)
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 10000 bits/sec, 17 packets/sec
8709 packets input, 845553 bytes, 0 no buffer
Received 745 broadcasts, 0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
0 input packets with dribble condition detected
18652077 packets output, 1480488712 bytes, 0 underruns
0 output errors, 0 collisions, 1 interface resets
0 babbles, 0 late collision, 0 deferred
0 lost carrier, 0 no carrier
0 output buffer failures, 0 output buffers swapped out

The following is sample output from the `show firewall multiple-vlan-interfaces` command:

Router# `show firewall multiple-vlan-interfaces`
Multiple firewall vlan interfaces feature is enabled

The following is sample output from the `show firewall module` command:

Router# `show firewall module`
Module Vlan-groups
  5  50,52
  8  51,52

The following is sample output from the `show firewall module [mod-num] version` command:

Router# `show firewall module 2 version`
ASA Service Module 2:
Sw Version: 100.7(8)19
The following is sample output from the `show firewall vlan-group` command:

```
Router# show firewall vlan-group
Group vlans
----- ------
  50 55-57
  51 70-85
  52 100
```

The following is sample output from the `show interface vlan` command:

```
Router# show interface vlan 55
Vlan55 is up, line protocol is up
Hardware is EtherSVI, address is 0008.20de.45ca (bia 0008.20de.45ca)
Internet address is 10.1.1.1/24
MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
    reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA, loopback not set
ARP type:ARPA, ARP Timeout 04:00:00
Last input never, output 00:00:08, output hang never
Last clearing of "show interface" counters never
Queueing strategy:fifo
Output queue :0/40 (size/max)
    5 minute input rate  0 bits/sec,  0 packets/sec
    5 minute output rate 0 bits/sec,  0 packets/sec
L2 Switched:ucast:196 pkt, 13328 bytes - mcast:4 pkt, 256 bytes
L3 in Switched:ucast:0 pkt, 0 bytes - mcast:0 pkt, 0 bytes mcast
L3 out Switched:ucast:0 pkt, 0 bytes
    0 packets input, 0 bytes, 0 no buffer
    Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
    4 packets output, 256 bytes, 0 underruns
    0 output errors, 0 interface resets
    0 output buffer failures, 0 output buffers swapped out
```

### Feature History for the Switch for Use with the ASA Services Module

Table 2-1 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASA Services Module support on the Cisco Catalyst 6500 switch</td>
<td>8.5(1)</td>
<td>The ASASM is a high-performance security services module for the Catalyst 6500 series switch, which you configure according to the procedures in this chapter. We introduced or modified the following commands: firewall transparent, mac address auto, firewall autostate (IOS), interface vlan.</td>
</tr>
</tbody>
</table>
Getting Started

This chapter describes how to get started with your ASASM. This chapter includes the following sections:

- Accessing the ASA Services Module Command-Line Interface, page 3-1
- Configuring ASDM Access for the ASA Services Module, page 3-5
- Starting ASDM, page 3-7
- Working with the Configuration, page 3-11
- Applying Configuration Changes to Connections, page 3-16

Accessing the ASA Services Module Command-Line Interface

For initial configuration, access the command-line interface by connecting to the switch (either to the console port or remotely using Telnet or SSH) and then connecting to the ASASM. This section describes how to access the ASASM CLI, and includes the following sections:

- Logging Into the ASA Services Module, page 3-1
- Logging Out of a Console Session, page 3-3
- Logging Out of a Telnet Session, page 3-5

Logging Into the ASA Services Module

For initial configuration, access the command-line interface by connecting to the switch (either to the console port or remotely using Telnet or SSH) and then connecting to the ASASM.

If your system is already in multiple context mode, then accessing the ASASM from the switch places you in the system execution space. See Chapter 6, “Configuring Multiple Context Mode,” for more information about multiple context mode.

Later, you can configure remote access directly to the ASASM using Telnet or SSH according to the “Configuring ASA Access for ASDM, Telnet, or SSH” section on page 34-1.

This section includes the following topics:

- Information About Connection Methods, page 3-2
- Logging In, page 3-2
Information About Connection Methods

From the switch CLI, you can use two methods to connect to the ASASM:

- Telnet connection—Using the session command, you create a Telnet connection to the ASASM.
  
  Benefits include:
  - You can have multiple sessions to the ASASM at the same time.
  - The Telnet session is a fast connection.
  
  Limitations include:
  - The Telnet session is terminated when the ASASM reloads, and can time out.
  - You cannot access the ASASM until it completely loads; you cannot access ROMMON.

- Virtual console connection—Using the service-module session command, you create a virtual console connection to the ASASM, with all the benefits and limitations of an actual console connection.
  
  Benefits include:
  - The connection is persistent across reloads and does not time out.
  - You can stay connected through ASASM reloads and view startup messages.
  - You can access ROMMON if the ASASM cannot load the image.
  
  Limitations include:
  - The connection is slow (9600 baud).
  - You can only have one console connection active at a time.

Note

Because of the persistence of the connection, if you do not properly log out of the ASASM, the connection may exist longer than intended. If someone else wants to log in, they will need to kill the existing connection. See the “Logging Out of a Console Session” section on page 3-3 for more information.

Logging In

Perform the following steps to log into the ASASM and access global configuration mode.
Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>From the switch, perform one of the following:</td>
</tr>
<tr>
<td>`session [switch (1</td>
<td>2)] slot number processor 1`</td>
</tr>
<tr>
<td>You are prompted for the login password:</td>
<td>Note: The <code>session slot processor 0</code> command, which is supported on other services modules, is not supported on the ASASM; the ASASM does not have a processor 0.</td>
</tr>
<tr>
<td><code>hostname passwd:</code></td>
<td>To view the module slot numbers, enter the <code>show module</code> command at the switch prompt. Enter the login password to the ASASM. By default, the password is <code>cisco</code>. You access user EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>Router# session slot number processor 1</code></td>
<td></td>
</tr>
<tr>
<td><code>hostname passwd: cisco</code></td>
<td></td>
</tr>
<tr>
<td><code>hostname&gt;</code></td>
<td></td>
</tr>
<tr>
<td>`service-module session [switch (1</td>
<td>2)] slot number`</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>Router# service-module session slot 3</code></td>
<td></td>
</tr>
<tr>
<td><code>hostname&gt;</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>enable</td>
</tr>
<tr>
<td><code>hostname# enable</code></td>
<td>Accesses privileged EXEC mode, which is the highest privilege level. Enter the enable password at the prompt. By default, the password is blank. To change the enable password, see the “Configuring the Hostname, Domain Name, and Passwords” section on page 9-1. To exit privileged EXEC mode, enter the <code>disable</code>, <code>exit</code>, or <code>quit</code> command.</td>
</tr>
<tr>
<td><code>Password:</code></td>
<td></td>
</tr>
<tr>
<td><code>hostname#</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td><code>configure terminal</code></td>
</tr>
<tr>
<td><code>hostname# configure terminal</code></td>
<td>Accesses global configuration mode. To exit global configuration mode, enter the <code>disable</code>, <code>exit</code>, or <code>quit</code> command.</td>
</tr>
<tr>
<td><code>hostname(config)#</code></td>
<td></td>
</tr>
</tbody>
</table>

Logging Out of a Console Session

This section includes the following topics:

- Logging Out, page 3-4
- Killing an Active Console Connection, page 3-4
Logging Out

If you do not log out of the ASASM, the console connection persists; there is no timeout. To end the ASASM console session and access the switch CLI, perform the following steps.

To kill another user’s active connection, which may have been unintentionally left open, see the “Killing an Active Console Connection” section on page 3-4.

Detailed Steps

Step 1
To return to the switch CLI, type the following:

**Ctrl-Shift-6, x**

You return to the switch prompt:

```
asasm# [Ctrl-Shift-6, x]
Router#
```

**Note** Shift-6 on US and UK keyboards issues the caret (^) character. If you have a different keyboard and cannot issue the caret (^) character as a standalone character, you can temporarily or permanently change the escape character to a different character. Use the `terminal escape-character ascii_number` command (to change for this session) or the `default escape-character ascii_number` command (to change permanently). For example, to change the sequence for the current session to Ctrl-w, x, enter `terminal escape-character 23`.

Killing an Active Console Connection

Because of the persistence of a console connection, if you do not properly log out of the ASASM, the connection may exist longer than intended. If someone else wants to log in, they will need to kill the existing connection.

Detailed Steps

Step 1
From the switch CLI, show the connected users using the `show users` command. A console user is called “con”. The Host address shown is 127.0.0.slot0, where slot is the slot number of the module.

```
Router# show users
```

For example, the following command output shows a user “con” on line 0 on a module in slot 2:

```
Line       User       Host(s)              Idle       Location
*  0 con 0 127.0.0.20 00:00:02
```

Step 2
To clear the line with the console connection, enter the following command:

```
Router# clear line number
```

For example:

```
Router# clear line 0
```
Logging Out of a Telnet Session

To end the access the switch CLI and resume or disconnect the Telnet session, perform the following steps.

**Detailed Steps**

**Step 1**
To return to the switch CLI, type the following:

```
Ctrl-Shift-6, x
```

You return to the switch prompt:

```
asasm# [Ctrl-Shift-6, x]
Router#
```

---

**Note**
Shift-6 on US and UK keyboards issues the caret (^) character. If you have a different keyboard and cannot issue the caret (^) character as a standalone character, you can temporarily or permanently change the escape character to a different character. In Cisco IOS, before you session to the ASASM, use the `terminal escape-character ascii_number` command (to change temporarily) or the `default escape-character ascii_number` command (to change permanently). For example, to temporarily change the sequence to Ctrl-w, x, enter `terminal escape-character 23`. The next time you log into the switch, the escape character reverts back to the default.

**Step 2**
To resume your Telnet session, press the **Enter** key at the switch prompt.

**Step 3**
To disconnect your Telnet session, enter the following command at the switch CLI:

```
Router# disconnect
```

If you do not disconnect the session, it will eventually time out according to the ASASM configuration.

---

Configuring ASDM Access for the ASA Services Module

Because the ASASM does not have physical interfaces, it does not come pre-configured for ASDM access; you must configure ASDM access using the CLI on the ASASM. To configure the ASASM for ASDM access, perform the following steps.

**Prerequisites**

- Assign a VLAN interface to the ASASM according to the “Assigning VLANs to the ASA Services Module” section on page 2-4.
- Connect to the ASASM and access global configuration mode according to the “Accessing the ASA Services Module Command-Line Interface” section on page 3-1.
### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> (Optional) <strong>firewall transparent</strong></td>
<td>Enables transparent firewall mode. This command clears your configuration. See the “Configuring the Firewall Mode” section on page 5-1 for more information.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# firewall transparent</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Step 2</strong> Do one of the following to configure a management interface, depending on your mode:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Routed mode:</strong></td>
<td></td>
</tr>
<tr>
<td>interface vlan number  ip address ip_address [mask] nameif name security-level level</td>
<td>Configures an interface in routed mode. The security-level is a number between 1 and 100, where 100 is the most secure.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# interface vlan 1</td>
<td></td>
</tr>
<tr>
<td>hostname(config-if)# ip address 192.168.1.1 255.255.255.0</td>
<td></td>
</tr>
<tr>
<td>hostname(config-if)# nameif inside</td>
<td></td>
</tr>
<tr>
<td>hostname(config-if)# security-level 100</td>
<td></td>
</tr>
</tbody>
</table>

| **Transparent mode:**                             |                                                                         |
| interface bvi number  ip address ip_address [mask] |                                                                         |
| interface vlan number  bridge-group bvi_number nameif name security-level level | Configures a bridge virtual interface and assigns a management VLAN to the bridge group. The security-level is a number between 1 and 100, where 100 is the most secure. |
| **Example:**                                      |                                                                         |
| hostname(config)# interface bvi 1                  |                                                                         |
| hostname(config-if)# ip address 192.168.1.1 255.255.255.0 |                                                                         |
| hostname(config)# interface vlan 1                 |                                                                         |
| hostname(config-if)# bridge-group 1                |                                                                         |
| hostname(config-if)# nameif inside                 |                                                                         |
| hostname(config-if)# security-level 100            |                                                                         |

| **Step 3** dhcpd address ip_address-ip_address interface_name  dhcpd enable interface_name | Enables DHCP for the management host on the management interface network. Make sure you do not include the management address in the range. |
| **Example:**                                      |                                                                         |
| hostname(config)# dhcpd address 192.168.1.2-192.168.1.254 inside |                                                                         |
| hostname(config)# dhcpd enable inside              |                                                                         |

| **Step 4** http server enable                      | Enables the HTTP server for ASDM.                                       |
| **Example:**                                      |                                                                         |
| hostname(config)# http server enable               |                                                                         |
The following routed mode configuration configures the VLAN 1 interface and enables ASDM for a management host:

```plaintext
interface vlan 1
  nameif inside
  ip address 192.168.1.1 255.255.255.0
  security-level 100
dhcpd address 192.168.1.3-192.168.1.254 inside
dhcpd enable inside
http server enable
http 192.168.1.0 255.255.255.0 inside
```

The following configuration converts the firewall mode to transparent mode, configures the VLAN 1 interface and assigns it to BVI 1, and enables ASDM for a management host:

```plaintext
firewall transparent
interface bvi 1
  ip address 192.168.1.1 255.255.255.0
interface vlan 1
  bridge-group 1
  nameif inside
  security-level 100
dhcpd address 192.168.1.3-192.168.1.254 inside
dhcpd enable inside
http server enable
http 192.168.1.0 255.255.255.0 inside
```

**Starting ASDM**

You can start ASDM using two methods:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 5 http ip_address mask interface_name</td>
<td>Allows the management host to access ASDM.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostnames(config)# http 192.168.1.0 255.255.255.0 management</td>
</tr>
<tr>
<td>Step 6 write memory</td>
<td>Saves the configuration.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostnames(config)# write memory</td>
</tr>
<tr>
<td>Step 7 (Optional) mode multiple</td>
<td>Sets the mode to multiple mode. When prompted, confirm that you want to convert the existing configuration to be the admin context. You are then prompted to reload the ASASM. See Chapter 6, “Configuring Multiple Context Mode,” for more information.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostnames(config)# mode multiple</td>
</tr>
<tr>
<td>Step 8</td>
<td>To launch ASDM, see the “Starting ASDM” section on page 3-7.</td>
</tr>
</tbody>
</table>

Examples
Starting ASDM

- **ASDM-IDM Launcher (Windows only)—**The Launcher is an application downloaded from the ASASM using a web browser that you can use to connect to any ASASM IP address. You do not need to re-download the launcher if you want to connect to other ASASMs. The Launcher also lets you run a virtual ASDM in Demo mode using files downloaded locally.

- **Java Web Start—**For each ASASM that you manage, you need to connect with a web browser and then save or launch the Java Web Start application. You can optionally save the application to your PC; however you need separate applications for each ASASM IP address.

**Note**

Within ASDM, you can choose a different ASASM IP address to manage; the difference between the Launcher and Java Web Start application functionality rests primarily in how you initially connect to the ASASM and launch ASDM.

### This section describes how to connect to ASDM initially, and then launch ASDM using the Launcher or the Java Web Start application. This section includes the following topics:

- Connecting to ASDM for the First Time, page 3-8
- Starting ASDM from the ASDM-IDM Launcher, page 3-9
- Starting ASDM from the Java Web Start Application, page 3-9
- Using ASDM in Demo Mode, page 3-10

**Note**

ASDM allows multiple PCs or workstations to each have one browser session open with the same ASASM software. A single ASASM can support up to five concurrent ASDM sessions in single, routed mode. Only one session per browser per PC or workstation is supported for a specified ASASM. In multiple context mode, five concurrent ASDM sessions are supported per context, up to a maximum of 32 total connections for each ASASM.

### Connecting to ASDM for the First Time

To connect to ASDM for the first time to download the ASDM-IDM Launcher or Java Web Start application, perform the following steps:

**Step 1**

From a supported web browser on the ASASM network, enter the following URL:

https://interface_ip_address/admin

Where *interface_ip_address* is the management IP address of the ASASM. See the “Configuring ASDM Access for the ASA Services Module” section on page 3-5 for more information about management access.

See the ASDM release notes for your release for the requirements to run ASDM.

The ASDM launch page appears with the following buttons:

- **Install ASDM Launcher and Run ASDM** (Windows only)
- **Run ASDM**
- **Run Startup Wizard**

**Step 2**

To download the Launcher:

a. Click **Install ASDM Launcher and Run ASDM**.
b. Enter the username and password, and click **OK**. For a factory default configuration, leave these fields empty. With no HTTPS authentication configured, you can gain access to ASDM with no username and the **enable** password, which is blank by default. With HTTPS authentication enabled, enter your username and associated password.

c. Save the installer to your PC, and then start the installer. The ASDM-IDM Launcher opens automatically after installation is complete.

d. See the “Starting ASDM from the ASDM-IDM Launcher” section on page 3-9 to use the Launcher to connect to ASDM.

**Step 3**

To use the Java Web Start application:

a. Click **Run ASDM** or **Run Startup Wizard**.

b. Save the application to your PC when prompted. You can optionally open it instead of saving it.

c. See the “Starting ASDM from the Java Web Start Application” section on page 3-9 to use the Java Web Start application to connect to ASDM.

---

### Starting ASDM from the ASDM-IDM Launcher

To start ASDM from the ASDM-IDM Launcher, perform the following steps.

**Prerequisites**

Download the ASDM-IDM Launcher according to the “Connecting to ASDM for the First Time” section on page 3-8.

**Detailed Steps**

**Step 1**

Start the ASDM-IDM Launcher application.

**Step 2**

Enter or choose the ASASM IP address or hostname to which you want to connect. To clear the list of IP addresses, click the trash can icon next to the Device/IP Address/Name field.

**Step 3**

Enter your username and your password, and then click **OK**.

For a factory default configuration, leave these fields empty. With no HTTPS authentication configured, you can gain access to ASDM with no username and the **enable** password, which is blank by default. With HTTPS authentication enabled, enter your username and associated password.

If there is a new version of ASDM on the ASASM, the ASDM Launcher automatically downloads the new version and requests that you update the current version before starting ASDM.

The main ASDM window appears.

---

### Starting ASDM from the Java Web Start Application

To start ASDM from the Java Web Start application, perform the following steps.
Starting ASDM

Chapter 3    Getting Started

Prerequisites

Download the Java Web Start application according to the “Connecting to ASDM for the First Time” section on page 3-8.

Detailed Steps

- **Step 1** Start the Java Web Start application.
- **Step 2** Accept any certificates according to the dialog boxes that appear. The Cisco ASDM-IDM Launcher appears.
- **Step 3** Enter the username and password, and click **OK**. For a factory default configuration, leave these fields empty. With no HTTPS authentication configured, you can gain access to ASDM with no username and the **enable** password, which is blank by default. With HTTPS authentication enabled, enter your username and associated password.

The main ASDM window appears.

Using ASDM in Demo Mode

The ASDM Demo Mode, a separately installed application, lets you run ASDM without having a live device available. In this mode, you can do the following:

- Perform configuration and selected monitoring tasks via ASDM as though you were interacting with a real device.
- Demonstrate ASDM or ASASM features using the ASDM interface.
- Perform configuration and monitoring tasks with the CSC SSM.
- Obtain simulated monitoring and logging data, including real-time syslog messages. The data shown is randomly generated; however, the experience is identical to what you would see when you are connected to a real device.

This mode has been updated to support the following features:

- For global policies, an ASASM in single, routed mode and intrusion prevention
- For object NAT, an ASASM in single, routed mode and a firewall DMZ.
- For the Botnet Traffic Filter, an ASASM in single, routed mode and security contexts.
- Site-to-Site VPN with IPv6 (Clientless SSL VPN and IPsec VPN)
- Promiscuous IDS (intrusion prevention)
- Unified Communication Wizard

This mode does not support the following:

- Saving changes made to the configuration that appear in the GUI.
- File or disk operations.
- Historical monitoring data.
- Non-administrative users.
- These features:
  - File menu:
Chapter 3      Getting Started

Working with the Configuration

This section describes how to work with the configuration. The ASASM loads the configuration from a text file, called the startup configuration. This file resides by default as a hidden file in internal flash memory. You can, however, specify a different path for the startup configuration. (For more information, see Chapter 56, “Managing Software and Configurations.”)

Save Running Configuration to Flash
Save Running Configuration to TFTP Server
Save Running Configuration to Standby Unit
Save Internal Log Buffer to Flash
Clear Internal Log Buffer

- Tools menu:
  Command Line Interface
  Ping
  File Management
  Update Software
  File Transfer
  Upload Image from Local PC
  System Reload

- Toolbar/Status bar > Save
- Configuration > Interface > Edit Interface > Renew DHCP Lease
- Configuring a standby device after failover

- Operations that cause a rereading of the configuration, in which the GUI reverts to the original configuration:
  - Switching contexts
  - Making changes in the Interface pane
  - NAT pane changes
  - Clock pane changes

To run ASDM in Demo Mode, perform the following steps:

**Step 1** Download the ASDM Demo Mode installer, asdm-demo-version.msi, from the following location: http://www.cisco.com/cisco/web/download/index.html.

**Step 2** Double-click the installer to install the software.

**Step 3** Double-click the Cisco ASDM Launcher shortcut on your desktop, or open it from the Start menu.

**Step 4** Check the Run in Demo Mode check box.

The Demo Mode window appears.
When you enter a command, the change is made only to the running configuration in memory. You must manually save the running configuration to the startup configuration for your changes to remain after a reboot.

The information in this section applies to both single and multiple security contexts, except where noted. Additional information about contexts is in Chapter 6, “Configuring Multiple Context Mode.”

This section includes the following topics:

- Saving Configuration Changes, page 3-12
- Copying the Startup Configuration to the Running Configuration, page 3-14
- Viewing the Configuration, page 3-14
- Clearing and Removing Configuration Settings, page 3-15
- Creating Text Configuration Files Offline, page 3-15

## Saving Configuration Changes

This section describes how to save your configuration and includes the following topics:

- Saving Configuration Changes in Single Context Mode, page 3-12
- Saving Configuration Changes in Multiple Context Mode, page 3-12

### Saving Configuration Changes in Single Context Mode

To save the running configuration to the startup configuration, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>write memory</td>
<td>Saves the running configuration to the startup configuration.</td>
</tr>
</tbody>
</table>

**Note**  
The `copy running-config startup-config` command is equivalent to the `write memory` command.

**Example:**

```
hostname# write memory
```

### Saving Configuration Changes in Multiple Context Mode

You can save each context (and system) configuration separately, or you can save all context configurations at the same time. This section includes the following topics:

- Saving Each Context and System Separately, page 3-12
- Saving All Context Configurations at the Same Time, page 3-13

### Saving Each Context and System Separately

To save the system or context configuration, enter the following command within the system or context:
Chapter 3      Getting Started

Working with the Configuration

Saving All Context Configurations at the Same Time

To save all context configurations at the same time, as well as the system configuration, enter the following command in the system execution space:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>write memory</td>
<td>Saves the running configuration to the startup configuration. For multiple context mode, context startup configurations can reside on external servers. In this case, the ASASM saves the configuration back to the server you identified in the context URL, except for an HTTP or HTTPS URL, which do not let you save the configuration to the server.</td>
</tr>
<tr>
<td>Example: hostname# write memory</td>
<td></td>
</tr>
</tbody>
</table>

Note The copy running-config startup-config command is equivalent to the write memory command.

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>write memory all [/noconfirm]</td>
<td>Saves the running configuration to the startup configuration for all contexts and the system configuration. If you do not enter the /noconfirm keyword, you see the following prompt: Are you sure [Y/N]: After you enter Y, the ASASM saves the system configuration and each context. Context startup configurations can reside on external servers. In this case, the ASASM saves the configuration back to the server you identified in the context URL, except for an HTTP or HTTPS URL, which do not let you save the configuration to the server.</td>
</tr>
<tr>
<td>Example: hostname# write memory all /noconfirm</td>
<td></td>
</tr>
</tbody>
</table>

After the ASASM saves each context, the following message appears:

'Saving context 'b' ... ( 1/3 contexts saved ) '

Sometimes, a context is not saved because of an error. See the following information for errors:

- For contexts that are not saved because of low memory, the following message appears:

  The context 'context a' could not be saved due to Unavailability of resources

- For contexts that are not saved because the remote destination is unreachable, the following message appears:

  The context 'context a' could not be saved due to non-reachability of destination

- For contexts that are not saved because the context is locked, the following message appears:

  Unable to save the configuration for the following contexts as these contexts are locked.
  context 'a', context 'x', context 'z'.

A context is only locked if another user is already saving the configuration or in the process of deleting the context.

- For contexts that are not saved because the startup configuration is read-only (for example, on an HTTP server), the following message report is printed at the end of all other messages:
Unable to save the configuration for the following contexts as these contexts have read-only config-urls:
context ‘a’ , context ‘b’ , context ‘c’.

- For contexts that are not saved because of bad sectors in the flash memory, the following message appears:
The context ‘context a’ could not be saved due to Unknown errors

### Copying the Startup Configuration to the Running Configuration

Copy a new startup configuration to the running configuration using one of the following options.

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>copy startup-config running-config</td>
<td>Merges the startup configuration with the running configuration. A merge adds any new commands from the new configuration to the running configuration. If the configurations are the same, no changes occur. If commands conflict or if commands affect the running of the context, then the effect of the merge depends on the command. You might get errors, or you might have unexpected results.</td>
</tr>
<tr>
<td>reload</td>
<td>Reloads the ASASM, which loads the startup configuration and discards the running configuration.</td>
</tr>
<tr>
<td>clear configure all copy startup-config running-config</td>
<td>Loads the startup configuration and discards the running configuration without requiring a reload.</td>
</tr>
</tbody>
</table>

### Viewing the Configuration

The following commands let you view the running and startup configurations.

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show running-config</td>
<td>Views the running configuration.</td>
</tr>
<tr>
<td>show running-config command</td>
<td>Views the running configuration of a specific command.</td>
</tr>
<tr>
<td>show startup-config</td>
<td>Views the startup configuration.</td>
</tr>
</tbody>
</table>
Clearing and Removing Configuration Settings

To erase settings, enter one of the following commands.

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>clear configure configuration command [level2configurationcommand]</td>
<td>Clears all the configuration for a specified command. If you only want to clear the configuration for a specific version of the command, you can enter a value for level2configurationcommand. For example, to clear the configuration for all aaa commands, enter the following command: hostname(config)# clear configure aaa To clear the configuration for only aaa authentication commands, enter the following command: hostname(config)# clear configure aaa authentication</td>
</tr>
<tr>
<td>no configuration command [level2configurationcommand] qualifier</td>
<td>Disables the specific parameters or options of a command. In this case, you use the no command to remove the specific configuration identified by qualifier. For example, to remove a specific nat command, enter enough of the command to identify it uniquely as follows: hostname(config)# no nat (inside) 1</td>
</tr>
<tr>
<td>write erase</td>
<td>Erases the startup configuration.</td>
</tr>
<tr>
<td>clear configure all</td>
<td>Erases the running configuration. Note In multiple context mode, if you enter clear configure all from the system configuration, you also remove all contexts and stop them from running. The context configuration files are not erased, and remain in their original location.</td>
</tr>
</tbody>
</table>

Creating Text Configuration Files Offline

This guide describes how to use the CLI to configure the ASASM; when you save commands, the changes are written to a text file. Instead of using the CLI, however, you can edit a text file directly on your PC and paste a configuration at the configuration mode command-line prompt in its entirety, or line by line. Alternatively, you can download a text file to the ASASM internal flash memory. See Chapter 56, “Managing Software and Configurations,” for information on downloading the configuration file to the ASASM.

In most cases, commands described in this guide are preceded by a CLI prompt. The prompt in the following example is “hostname(config)#”:

hostname(config)# context a

In the text configuration file you are not prompted to enter commands, so the prompt is omitted as follows:

context a
For additional information about formatting the file, see Appendix A, “Using the Command-Line Interface.”

### Applying Configuration Changes to Connections

When you make security policy changes to the configuration, all *new* connections use the new security policy. Existing connections continue to use the policy that was configured at the time of the connection establishment. To ensure that all connections use the new policy, you need to disconnect the current connections so they can reconnect using the new policy. To disconnect connections, enter one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>clear local-host [ip_address] [all]</td>
<td>This command reinitializes per-client run-time states such as connection limits and embryonic limits. As a result, this command removes any connection that uses those limits. See the <code>show local-host all</code> command to view all current connections per host. With no arguments, this command clears all affected through-the-box connections. To also clear to-the-box connections (including your current management session), use the <code>all</code> keyword. To clear connections to and from a particular IP address, use the <code>ip_address</code> argument.</td>
</tr>
<tr>
<td>clear conn [all] [protocol {tcp</td>
<td>udp}] [address src_ip[-src_ip] [netmask mask]] [port src_port[-src_port]] [address dest_ip[-dest_ip] [netmask mask]] [port dest_port[-dest_port]]</td>
</tr>
<tr>
<td>clear xlate [arguments]</td>
<td>This command clears dynamic NAT sessions; static sessions are not affected. As a result, it removes any connections using those NAT sessions. With no arguments, this command clears all NAT sessions. See the command reference for more information about the arguments available.</td>
</tr>
</tbody>
</table>

**Examples:**

```plaintext
hostname(config)# clear local-host all
```

```plaintext
hostname(config)# clear conn all
```

```plaintext
hostname(config)# clear xlate
```

```plaintext
hostname(config)# clear xlate
```
Managing Feature Licenses

A license specifies the options that are enabled on a given ASASM. This document describes how to obtain a license activation key and how to activate it. It also describes the available licenses for each model.

Note

This chapter describes licensing for Version 8.5; for other versions, see the licensing documentation that applies to your version:


This chapter includes the following sections:

- Supported Feature Licenses, page 4-1
- Information About Feature Licenses, page 4-3
- Guidelines and Limitations, page 4-8
- Configuring Licenses, page 4-9
- Monitoring Licenses, page 4-11
- Feature History for Licensing, page 4-13

Supported Feature Licenses

This section describes the licenses available as well as important notes about licenses. This section includes the following topics:

- Licenses, page 4-1
- License Notes, page 4-3

Licenses

Items that are in italics are separate, optional licenses with which that you can replace the Base license. You can mix and match licenses. For detailed information about licenses, see the “License Notes” section on page 4-3.
ASA Services Module

Table 4-1 shows the licenses for the ASASM. All ASASM licenses for this release are No Payload Encryption licenses. See the “No Payload Encryption Models” section on page 4-7 for more information.

### Table 4-1 ASASM License Features

<table>
<thead>
<tr>
<th>Licenses</th>
<th>Description (Base License in Plain Text)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Firewall Licenses</td>
<td></td>
</tr>
<tr>
<td>Botnet Traffic Filter(^1)</td>
<td>Disabled</td>
</tr>
<tr>
<td>Firewall Conns, Concurrent</td>
<td>8,000,000</td>
</tr>
<tr>
<td>GTP/GPRS</td>
<td>Disabled</td>
</tr>
<tr>
<td>Intercompany Media Engine</td>
<td>No support.</td>
</tr>
<tr>
<td>UC Phone Proxy Sessions</td>
<td>No support.</td>
</tr>
<tr>
<td>VPN Licenses</td>
<td></td>
</tr>
<tr>
<td>Adv. Endpoint Assessment</td>
<td>No support.</td>
</tr>
<tr>
<td>AnyConnect for Cisco VPN Phone</td>
<td>No support.</td>
</tr>
<tr>
<td>AnyConnect Essentials</td>
<td>No support.</td>
</tr>
<tr>
<td>AnyConnect for Mobile</td>
<td>No support.</td>
</tr>
<tr>
<td>AnyConnect Premium (sessions)</td>
<td>No support.</td>
</tr>
<tr>
<td>Total VPN (sessions), combined all types</td>
<td>No support.</td>
</tr>
<tr>
<td>Other VPN (sessions)</td>
<td>No support.</td>
</tr>
<tr>
<td>VPN Load Balancing</td>
<td>No support.</td>
</tr>
<tr>
<td>General Licenses</td>
<td></td>
</tr>
<tr>
<td>Encryption</td>
<td>Base (DES)</td>
</tr>
<tr>
<td></td>
<td>Optional license: Strong (3DES/AES)</td>
</tr>
<tr>
<td>Failover</td>
<td>Active/Standby or Active/Active</td>
</tr>
<tr>
<td>Interfaces of all types, Max.(^1)</td>
<td>4128</td>
</tr>
<tr>
<td>Security Contexts</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Optional licenses:</td>
</tr>
<tr>
<td></td>
<td>5  10  20  50  100  250</td>
</tr>
<tr>
<td>VLANs, Maximum</td>
<td>1000</td>
</tr>
</tbody>
</table>

\(^1\) See the “License Notes” section on page 4-3.
License Notes

Table 4-2 includes common footnotes shared by multiple tables in the “Licenses” section on page 4-1.

<table>
<thead>
<tr>
<th>License</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Botnet Traffic Filter</td>
<td>Requires a Strong Encryption (3DES/AES) License to download the dynamic database.</td>
</tr>
<tr>
<td>Encryption</td>
<td>The DES license cannot be disabled. If you have the 3DES license installed, DES is still available. To prevent the use of DES when you want to only use strong encryption, be sure to configure any relevant commands to use only string encryption.</td>
</tr>
<tr>
<td>Failover, Active/Active</td>
<td>You cannot use Active/Active failover and VPN; if you want to use VPN, use Active/Standby failover.</td>
</tr>
<tr>
<td>Interfaces of all types, Max.</td>
<td>The maximum number of combined interfaces; for example, VLANs, physical, redundant, bridge group, and EtherChannel interfaces.</td>
</tr>
</tbody>
</table>

Information About Feature Licenses

A license specifies the options that are enabled on a given ASASM. It is represented by an activation key that is a 160-bit (5 32-bit words or 20 bytes) value. This value encodes the serial number (an 11 character string) and the enabled features.

This section includes the following topics:

- Preinstalled License, page 4-3
- Permanent License, page 4-3
- Time-Based Licenses, page 4-4
- Failover Licenses, page 4-6
- No Payload Encryption Models, page 4-7
- Licenses FAQ, page 4-8

Preinstalled License

By default, your ASASM ships with a license already installed. This license might be the Base License, to which you want to add more licenses, or it might already have all of your licenses installed, depending on what you ordered and what your vendor installed for you. See the “Monitoring Licenses” section on page 4-11 section to determine which licenses you have installed.

Permanent License

You can have one permanent activation key installed. The permanent activation key includes all licensed features in a single key. If you also install time-based licenses, the ASASM combines the permanent and time-based licenses into a running license. See the “How Permanent and Time-Based Licenses Combine” section on page 4-4 for more information about how the ASASM combines the licenses.
Time-Based Licenses

In addition to permanent licenses, you can purchase time-based licenses or receive an evaluation license that has a time-limit. For example, you might buy a Botnet Traffic Filter time-based license that is valid for 1 year.

This section includes the following topics:

- Time-Based License Activation Guidelines, page 4-4
- How the Time-Based License Timer Works, page 4-4
- How Permanent and Time-Based Licenses Combine, page 4-4
- Stacking Time-Based Licenses, page 4-5
- Time-Based License Expiration, page 4-5

Time-Based License Activation Guidelines

- You can install multiple time-based licenses, including multiple licenses for the same feature. However, only one time-based license per feature can be active at a time. The inactive license remains installed, and ready for use.
- If you activate an evaluation license that has multiple features in the key, then you cannot also activate another time-based license for one of the included features.

How the Time-Based License Timer Works

- The timer for the time-based license starts counting down when you activate it on the ASASM.
- If you stop using the time-based license before it times out, then the timer halts. The timer only starts again when you reactivate the time-based license.
- If the time-based license is active, and you shut down the ASASM, then the timer continues to count down. If you intend to leave the ASASM in a shut down state for an extended period of time, then you should deactivate the time-based license before you shut down.

Note
We suggest you do not change the system clock after you install the time-based license. If you set the clock to be a later date, then if you reload, the ASASM checks the system clock against the original installation time, and assumes that more time has passed than has actually been used. If you set the clock back, and the actual running time is greater than the time between the original installation time and the system clock, then the license immediately expires after a reload.

How Permanent and Time-Based Licenses Combine

When you activate a time-based license, then features from both permanent and time-based licenses combine to form the running license. How the permanent and time-based licenses combine depends on the type of license. Table 4-3 lists the combination rules for each feature license.

Note
Even when the permanent license is used, if the time-based license is active, it continues to count down.
Stacking Time-Based Licenses

In many cases, you might need to renew your time-based license and have a seamless transition from the old license to the new one. For features that are only available with a time-based license, it is especially important that the license not expire before you can apply the new license. The ASASM allows you to stack time-based licenses so you do not have to worry about the license expiring or about losing time on your licenses because you installed the new one early.

When you install an identical time-based license as one already installed, then the licenses are combined, and the duration equals the combined duration.

For example:

1. You install a 52-week Botnet Traffic Filter license, and use the license for 25 weeks (27 weeks remain).
2. You then purchase another 52-week Botnet Traffic Filter license. When you install the second license, the licenses combine to have a duration of 79 weeks (52 weeks plus 27 weeks).

If the licenses are not identical, then the licenses are not combined. Because only one time-based license per feature can be active, only one of the licenses can be active. See the “Activating or Deactivating Keys” section on page 4-10 for more information about activating licenses.

Although non-identical licenses do not combine, when the current license expires, the ASASM automatically activates an installed license of the same feature if available. See the “Time-Based License Expiration” section on page 4-5 for more information.

Time-Based License Expiration

When the current license for a feature expires, the ASASM automatically activates an installed license of the same feature if available. If there are no other time-based licenses available for the feature, then the permanent license is used.
If you have more than one additional time-based license installed for a feature, then the ASASM uses the first license it finds; which license is used is not user-configurable and depends on internal operations. If you prefer to use a different time-based license than the one the ASASM activated, then you must manually activate the license you prefer. See the “Activating or Deactivating Keys” section on page 4-10.

**Failover Licenses**

With some exceptions, failover units do not require the same license on each unit. This section includes the following topics:

- Failover License Requirements and Exceptions, page 4-6
- How Failover Licenses Combine, page 4-6
- Loss of Communication Between Failover Units, page 4-7
- Upgrading Failover Pairs, page 4-7

**Failover License Requirements and Exceptions**

Failover units do not require the same license on each unit.

The exceptions to this rule include:

- Encryption license—Both units must have the same encryption license.

**Note**

A valid permanent key is required; in rare instances, your authentication key can be removed. If your key consists of all 0’s, then you need to reinstall a valid authentication key before failover can be enabled.

**How Failover Licenses Combine**

For failover pairs, the licenses on each unit are combined into a single running failover cluster license. For Active/Active failover, the license usage of the two units combined cannot exceed the failover cluster license.

If you buy separate licenses for the primary and secondary unit, then the combined license uses the following rules:

- For licenses that have numerical tiers, such as the number of sessions, the values from both the primary and secondary licenses are combined up to the platform limit. If both licenses in use are time-based, then the licenses count down simultaneously.
  
  For example:
  - You have two ASA 5540s, one with 20 contexts and the other with 10 contexts; the combined license allows 30 contexts. For Active/Active failover, one unit can use 18 contexts and the other unit can use 12 contexts, for example, for a total of 30; the combined usage cannot exceed the failover cluster license (in this case, 30).
  - For licenses that have a status of enabled or disabled, then the license with the enabled status is used.
For time-based licenses that are enabled or disabled (and do not have numerical tiers), the duration is the combined duration of both licenses. The primary unit counts down its license first, and when it expires, the secondary unit starts counting down its license. This rule also applies to Active/Active failover, even though both units are actively operating.

For example, if you have 48 weeks left on the Botnet Traffic Filter license on both units, then the combined duration is 96 weeks.

To view the combined license, see the “Monitoring Licenses” section on page 4-11.

**Loss of Communication Between Failover Units**

If the failover units lose communication for more than 30 days, then each unit reverts to the license installed locally. During the 30-day grace period, the combined running license continues to be used by both units.

If you restore communication during the 30-day grace period, then for time-based licenses, the time elapsed is subtracted from the primary license; if the primary license becomes expired, only then does the secondary license start to count down.

If you do not restore communication during the 30-day period, then for time-based licenses, time is subtracted from both primary and secondary licenses, if installed. They are treated as two separate licenses and do not benefit from the failover combined license. The time elapsed includes the 30-day grace period.

For example:

1. You have a 52-week Botnet Traffic Filter license installed on both units. The combined running license allows a total duration of 104 weeks.
2. The units operate as a failover unit for 10 weeks, leaving 94 weeks on the combined license (42 weeks on the primary, and 52 weeks on the secondary).
3. If the units lose communication (for example the primary unit fails over to the secondary unit), the secondary unit continues to use the combined license, and continues to count down from 94 weeks.
4. The time-based license behavior depends on when communication is restored:
   - Within 30 days—The time elapsed is subtracted from the primary unit license. In this case, communication is restored after 4 weeks. Therefore, 4 weeks are subtracted from the primary license leaving 90 weeks combined (38 weeks on the primary, and 52 weeks on the secondary).
   - After 30 days—The time elapsed is subtracted from both units. In this case, communication is restored after 6 weeks. Therefore, 6 weeks are subtracted from both the primary and secondary licenses, leaving 84 weeks combined (36 weeks on the primary, and 46 weeks on the secondary).

**Upgrading Failover Pairs**

Because failover pairs do not require the same license on both units, you can apply new licenses to each unit without any downtime. If you apply a permanent license that requires a reload (see Table 4-4 on page 4-10), then you can fail over to the other unit while you reload. If both units require reloading, then you can reload them separately so you have no downtime.

**No Payload Encryption Models**

The ASASM is only available as a No Payload Encryption model for this release. The ASASM software senses a No Payload Encryption model, and disables the following features:
• Unified Communications
• VPN

You can still install the Strong Encryption (3DES/AES) license for use with management connections. For example, you can use ASDM HTTPS/SSL, SSHv2, Telnet and SNMPv3. You can also download the dynamic database for the Botnet Traffic Filer (which uses SSL).

When you view the license (see the “Monitoring Licenses” section on page 4-11), VPN and Unified Communications licenses will not be listed.

**Licenses FAQ**

**Q.** Can I activate multiple time-based licenses?

**A.** Yes. You can use one time-based license per feature at a time.

**Q.** Can I “stack” time-based licenses so that when the time limit runs out, it will automatically use the next license?

**A.** Yes. For identical licenses, the time limit is combined when you install multiple time-based licenses. For non-identical licenses, the ASASM automatically activates the next time-based license it finds for the feature.

**Q.** Can I install a new permanent license while maintaining an active time-based license?

**A.** Yes. Activating a permanent license does not affect time-based licenses.

**Q.** Do I need to buy the same licenses for the secondary unit in a failover pair?

**A.** No, you do not have to have matching licenses on both units. Typically, you buy a license only for the primary unit; the secondary unit inherits the primary license when it becomes active. In the case where you also have a separate license on the secondary unit, the licenses are combined into a running failover cluster license, up to the model limits.

**Guidelines and Limitations**

See the following guidelines for activation keys.

**Context Mode Guidelines**

• In multiple context mode, apply the activation key in the system execution space.

**Firewall Mode Guidelines**

All license types are available in both routed and transparent mode.

**Failover Guidelines**

• Failover units do not require the same license on each unit.

Older versions of ASASM software required that the licenses match on each unit. Starting with Version 8.3(1), you no longer need to install identical licenses. Typically, you buy a license only for the primary unit; for Active/Standby failover, the secondary unit inherits the primary license when it becomes active. If you have licenses on both units, they combine into a single running failover cluster license.
Additional Guidelines and Limitations

- The activation key is not stored in your configuration file; it is stored as a hidden file in flash memory.
- The activation key is tied to the serial number of the device. Feature licenses cannot be transferred between devices (except in the case of a hardware failure). If you have to replace your device due to a hardware failure and it is covered by Cisco TAC, contact the Cisco Licensing Team to have your existing license transferred to the new serial number. The Cisco Licensing Team will ask for the Product Authorization Key reference number and existing serial number.
- Once purchased, you cannot return a license for a refund or for an upgraded license.

Configuring Licenses

This section includes the following topics:

- Obtaining an Activation Key, page 4-9
- Activating or Deactivating Keys, page 4-10

Obtaining an Activation Key

To obtain an activation key, you need a Product Authorization Key, which you can purchase from your Cisco account representative. You need to purchase a separate Product Activation Key for each feature license.

After obtaining the Product Authorization Keys, register them on Cisco.com by performing the following steps.

Detailed Steps

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
</table>
| Step 1 | Obtain the serial number for your ASASM by entering the following command.  
hostname# show activation-key |
| Step 2 | If you are not already registered with Cisco.com, create an account. |
| Step 3 | Go to the following licensing website:  
http://www.cisco.com/go/license |
| Step 4 | Enter the following information, when prompted:  
- Product Authorization Key (if you have multiple keys, enter one of the keys first. You have to enter each key as a separate process.)  
- The serial number of your ASASM  
- Your e-mail address  
An activation key is automatically generated and sent to the email address that you provide. This key includes all features you have registered so far for permanent licenses. For time-based licenses, each license has a separate activation key. |
| Step 5 | If you have additional Product Authorization Keys, repeat Step 4 for each Product Authorization Key. After you enter all of the Product Authorization Keys, the final activation key provided includes all of the permanent features you registered. |
Activating or Deactivating Keys

This section describes how to enter a new activation key, and how to activate and deactivate time-based keys.

Prerequisites

- If you are already in multiple context mode, enter the activation key in the system execution space.
- Some permanent licenses require you to reload the ASASM after you activate them. Table 4-4 lists the licenses that require reloading.

<table>
<thead>
<tr>
<th>Model</th>
<th>License Action Requiring Reload</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Changing the Encryption license.</td>
</tr>
<tr>
<td>All models</td>
<td>Downgrading any permanent license (for example, going from 10 contexts to 2 contexts).</td>
</tr>
</tbody>
</table>

Detailed Steps

**Command**

**Step 1**  
activation-key key [activate | deactivate]

Example:

```
hostname# activation-key 0xd11b3d48 0xa80a4c0a 0x48e0fd1c 0xb0443480 0x843fc490
```

Applies an activation key to the ASASM. The key is a five-element hexadecimal string with one space between each element. The leading 0x specifier is optional; all values are assumed to be hexadecimal.

You can install one permanent key, and multiple time-based keys. If you enter a new permanent key, it overwrites the already installed one.

The activate and deactivate keywords are available for time-based keys only. If you do not enter any value, activate is the default. The last time-based key that you activate for a given feature is the active one. To deactivate any active time-based key, enter the deactivate keyword. If you enter a key for the first time, and specify deactivate, then the key is installed on the ASASM in an inactive state. See the “Time-Based Licenses” section on page 4-4 for more information.

**Step 2**  
(Might be required.)

reload

Example:

```
hostname# reload
```

Reloads the ASASM. Some permanent licenses require you to reload the ASASM after entering the new activation key. See Table 4-4 on page 4-10 for a list of licenses that need reloading. If you need to reload, you will see the following message:

WARNING: The running activation key was not updated with the requested key. The flash activation key was updated with the requested key, and will become active after the next reload.
Monitoring Licenses

This section describes how to view your current license, and for time-based activation keys, how much time the license has left.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show activation-key [detail]</code></td>
<td>This command shows the permanent license, active time-based licenses, and the running license, which is a combination of the permanent license and active time-based licenses. The <code>detail</code> keyword also shows inactive time-based licenses. For failover units, this command also shows the “Failover cluster” license, which is the combined keys of the primary and secondary units.</td>
</tr>
</tbody>
</table>

Example: hostname# show activation-key detail

Example 4-1 Primary Unit Output for the ASA Services Module in a Failover Pair for show activation-key

The following is sample output from the `show activation-key` command for the primary failover unit that shows:

- The primary unit license (the combined permanent license and time-based licenses).
- The “Failover Cluster” license, which is the combined licenses from the primary and secondary units. This is the license that is actually running on the ASASM. The values in this license that reflect the combination of the primary and secondary licenses are in bold.
- The primary unit installed time-based licenses (active and inactive).

```
hostname# show activation-key

Serial Number: SAL144705BF
Running Permanent Activation Key: 0x4d1ed752 0xc8cfeb37 0xf4c38198 0x93c04c28 0x4a1c049a
Running Timebased Activation Key: 0xbc07bbd7 0xb15591e0 0xed68c013 0xd79374ff 0x44f87880

Licensed features for this platform:
Maximum Interfaces                : 1024           perpetual
Inside Hosts                      : Unlimited      perpetual
Failover                          : Active/Active  perpetual
DES                               : Enabled        perpetual
3DES-AES                          : Enabled        perpetual
Security Contexts                 : 25            perpetual
GTP/GPRS                          : Enabled        perpetual
Botnet Traffic Filter             : Enabled        330 days

This platform has an WS-SVC-ASA-SM1 No Payload Encryption license.

Failover cluster licensed features for this platform:
Maximum Interfaces                : 1024           perpetual
Inside Hosts                      : Unlimited      perpetual
Failover                          : Active/Active  perpetual
DES                               : Enabled        perpetual
3DES-AES                          : Enabled        perpetual
Security Contexts                 : 50             perpetual
GTP/GPRS                          : Enabled        perpetual
Botnet Traffic Filter             : Enabled        330 days
```
This platform has an WS-SVC-ASA-SM1 No Payload Encryption license.

The flash permanent activation key is the SAME as the running permanent key.

Active Timebased Activation Key:
0xbc07bbd7 0xb15591e0 0xed68c013 0xd79374ff 0x44f87880
Botnet Traffic Filter : Enabled 330 days

Example 4-2 Secondary Unit Output for the ASA Services Module in a Failover Pair for show activation-key

The following is sample output from the show activation-key command for the secondary failover unit that shows:

- The secondary unit license (the combined permanent license and time-based licenses).
- The “Failover Cluster” license, which is the combined licenses from the primary and secondary units. This is the license that is actually running on the ASASM. The values in this license that reflect the combination of the primary and secondary licenses are in bold.
- The secondary installed time-based licenses (active and inactive). This unit does not have any time-based licenses, so none display in this sample output.

hostname# show activation-key detail

Serial Number:  SAD143502E3
Running Permanent Activation Key: 0xf404c46a 0xb8e5bd84 0x28c1b900 0x92eca09c 0x4e2a0683

Licensed features for this platform:
Maximum Interfaces : 1024 perpetual
Inside Hosts : Unlimited perpetual
Failover : Active/Active perpetual
DES : Enabled perpetual
3DES-AES : Enabled perpetual
Security Contexts : 25 perpetual
GTP/GPRS : Disabled perpetual
Botnet Traffic Filter : Disabled perpetual

This platform has an WS-SVC-ASA-SM1 No Payload Encryption license.

Failover cluster licensed features for this platform:
Maximum Interfaces : 1024 perpetual
Inside Hosts : Unlimited perpetual
Failover : Active/Active perpetual
DES : Enabled perpetual
3DES-AES : Enabled perpetual
Security Contexts : 50 perpetual
GTP/GPRS : Enabled perpetual
Botnet Traffic Filter : Enabled 330 days

This platform has an WS-SVC-ASA-SM1 No Payload Encryption license.

The flash permanent activation key is the SAME as the running permanent key.
Feature History for Licensing

Table 4-5 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Increased Connections and VLANs</td>
<td>7.0(5)</td>
<td>Increased the following limits:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ASA5510 Base license connections from 32000 to 5000; VLANs from 0 to 10.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ASA5510 Security Plus license connections from 64000 to 130000; VLANs from 10 to 25.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ASA5520 connections from 130000 to 280000; VLANs from 25 to 100.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ASA5540 connections from 280000 to 400000; VLANs from 100 to 200.</td>
</tr>
<tr>
<td>SSL VPN Licenses</td>
<td>7.1(1)</td>
<td>SSL VPN licenses were introduced.</td>
</tr>
<tr>
<td>Increased SSL VPN Licenses</td>
<td>7.2(1)</td>
<td>A 5000-user SSL VPN license was introduced for the ASA 5550 and above.</td>
</tr>
<tr>
<td>Increased interfaces for the Base license on the</td>
<td>7.2(2)</td>
<td>For the Base license on the ASA 5510, the maximum number of interfaces was increased</td>
</tr>
<tr>
<td>ASA 5510</td>
<td></td>
<td>from 3 plus a management interface to unlimited interfaces.</td>
</tr>
<tr>
<td>Increased VLANs</td>
<td>7.2(2)</td>
<td>The maximum number of VLANs for the Security Plus license on the ASA 5505 was</td>
</tr>
<tr>
<td></td>
<td></td>
<td>increased from 5 (3 fully functional; 1 failover; one restricted to a backup</td>
</tr>
<tr>
<td></td>
<td></td>
<td>interface) to 20 fully functional interfaces. In addition, the number of trunk</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ports was increased from 1 to 8. Now there are 20</td>
</tr>
<tr>
<td></td>
<td></td>
<td>fully functional interfaces, you do not need to use the backup interface</td>
</tr>
<tr>
<td></td>
<td></td>
<td>command to cripple a backup ISP interface; you can use a fully-functional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>interface for it. The backup interface command is still useful for an Easy VPN</td>
</tr>
<tr>
<td></td>
<td></td>
<td>configuration. VLAN limits were also increased for the ASA 5510 (from 10 to 50 for</td>
</tr>
<tr>
<td></td>
<td></td>
<td>the Base license, and from 25 to 100 for the Security Plus license), the ASA 5520</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(from 100 to 150), the ASA 5550 (from 200 to 250).</td>
</tr>
<tr>
<td>Gigabit Ethernet Support for the ASA 5510</td>
<td>7.2(3)</td>
<td>The ASA 5510 now supports Gigabit Ethernet (1000 Mbps) for the Ethernet 0/0 and 0/1</td>
</tr>
<tr>
<td>Security Plus License</td>
<td></td>
<td>ports with the Security Plus license. In the Base license, they continue to be used</td>
</tr>
<tr>
<td></td>
<td></td>
<td>as Fast Ethernet (100 Mbps) ports. Ethernet 0/2, 0/3, and 0/4 remain as Fast</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Ethernet ports for both licenses.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note</strong> The interface names remain Ethernet 0/0 and Ethernet 0/1.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Use the <code>speed</code> command to change the speed on the interface and use the `show</td>
</tr>
<tr>
<td></td>
<td></td>
<td>interface` command to see what speed is currently configured for each interface.</td>
</tr>
</tbody>
</table>
### Table 4-5 Feature History for Licensing (continued)

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Advanced Endpoint Assessment License</td>
<td>8.0(2)</td>
<td>The Advanced Endpoint Assessment license was introduced. As a condition for the completion of a Cisco AnyConnect or clientless SSL VPN connections, the remote computer scans for a greatly expanded collection of antivirus and antispyware applications, firewalls, operating systems, and associated updates. It also scans for any registry entries, filenames, and process names that you specify. It sends the scan results to the ASASM. The ASASM uses both the user login credentials and the computer scan results to assign a Dynamic Access Policy (DAP). With an Advanced Endpoint Assessment License, you can enhance Host Scan by configuring an attempt to update noncompliant computers to meet version requirements. Cisco can provide timely updates to the list of applications and versions that Host Scan supports in a package that is separate from Cisco Secure Desktop.</td>
</tr>
<tr>
<td>VPN Load Balancing for the ASA 5510</td>
<td>8.0(2)</td>
<td>VPN load balancing is now supported on the ASA 5510 Security Plus license.</td>
</tr>
<tr>
<td>AnyConnect for Mobile License</td>
<td>8.0(3)</td>
<td>The AnyConnect for Mobile license was introduced. It lets Windows mobile devices connect to the ASASM using the AnyConnect client.</td>
</tr>
<tr>
<td>Time-based Licenses</td>
<td>8.0(4)/8.1(2)</td>
<td>Support for time-based licenses was introduced.</td>
</tr>
<tr>
<td>Increased VLANs for the ASA 5580</td>
<td>8.1(2)</td>
<td>The number of VLANs supported on the ASA 5580 are increased from 100 to 250.</td>
</tr>
<tr>
<td>Unified Communications Proxy Sessions</td>
<td>8.0(4)</td>
<td>The UC Proxy sessions license was introduced. Phone Proxy, Presence Federation Proxy, and Encrypted Voice Inspection applications use TLS proxy sessions for their connections. Each TLS proxy session is counted against the UC license limit. All of these applications are licensed under the UC Proxy umbrella, and can be mixed and matched. This feature is not available in Version 8.1.</td>
</tr>
<tr>
<td>Botnet Traffic Filter License</td>
<td>8.2(1)</td>
<td>The Botnet Traffic Filter license was introduced. The Botnet Traffic Filter protects against malware network activity by tracking connections to known bad domains and IP addresses.</td>
</tr>
</tbody>
</table>
### Feature History for Licensing (continued)

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
</table>
| AnyConnect Essentials License | 8.2(1) | The AnyConnect Essentials License was introduced. This license enables AnyConnect VPN client access to the ASASM. This license does not support browser-based SSL VPN access or Cisco Secure Desktop. For these features, activate an AnyConnect Premium license instead of the AnyConnect Essentials license.  
**Note** With the AnyConnect Essentials license, VPN users can use a Web browser to log in, and download and start (WebLaunch) the AnyConnect client. The AnyConnect client software offers the same set of client features, whether it is enabled by this license or an AnyConnect Premium license. The AnyConnect Essentials license cannot be active at the same time as the following licenses on a given ASASM: AnyConnect Premium license (all types) or the Advanced Endpoint Assessment license. You can, however, run AnyConnect Essentials and AnyConnect Premium licenses on different ASASMs in the same network. By default, the ASASM uses the AnyConnect Essentials license, but you can disable it to use other licenses by using the `no anyconnect-essentials` command. |
| SSL VPN license changed to AnyConnect Premium SSL VPN Edition license | 8.2(1) | The SSL VPN license name was changed to the AnyConnect Premium SSL VPN Edition license. |
| Shared Licenses for SSL VPN | 8.2(1) | Shared licenses for SSL VPN were introduced. Multiple ASASMs can share a pool of SSL VPN sessions on an as-needed basis. |
| Mobility Proxy application no longer requires Unified Communications Proxy license | 8.2(2) | The Mobility Proxy no longer requires the UC Proxy license. |
| 10 GE I/O license for the ASA 5585-X with SSP-20 | 8.2(3) | We introduced the 10 GE I/O license for the ASA 5585-X with SSP-20 to enable 10-Gigabit Ethernet speeds for the fiber ports. The SSP-60 supports 10-Gigabit Ethernet speeds by default.  
**Note** The ASA 5585-X is not supported in 8.3(x). |
| 10 GE I/O license for the ASA 5585-X with SSP-10 | 8.2(4) | We introduced the 10 GE I/O license for the ASA 5585-X with SSP-10 to enable 10-Gigabit Ethernet speeds for the fiber ports. The SSP-40 supports 10-Gigabit Ethernet speeds by default.  
**Note** The ASA 5585-X is not supported in 8.3(x). |
| Non-identical failover licenses | 8.3(1) | Failover licenses no longer need to be identical on each unit. The license used for both units is the combined license from the primary and secondary units. We modified the following commands: `show activation-key` and `show version`. |
Stackable time-based licenses | 8.3(1) | Time-based licenses are now stackable. In many cases, you might need to renew your time-based license and have a seamless transition from the old license to the new one. For features that are only available with a time-based license, it is especially important that the license not expire before you can apply the new license. The ASASM allows you to stack time-based licenses so you do not have to worry about the license expiring or about losing time on your licenses because you installed the new one early.

Intercompany Media Engine License | 8.3(1) | The IME license was introduced.

Multiple time-based licenses active at the same time | 8.3(1) | You can now install multiple time-based licenses, and have one license per feature active at a time. The following commands were modified: `show activation-key` and `show version`.

Discrete activation and deactivation of time-based licenses. | 8.3(1) | You can now activate or deactivate time-based licenses using a command. The following command was modified: `activation-key [activate | deactivate]`.

AnyConnect Premium SSL VPN Edition license changed to AnyConnect Premium SSL VPN license | 8.3(1) | The AnyConnect Premium SSL VPN Edition license name was changed to the AnyConnect Premium SSL VPN license.

No Payload Encryption image for export | 8.3(2) | If you install the No Payload Encryption software on the ASA 5505 through 5550, then you disable Unified Communications, strong encryption VPN, and strong encryption management protocols. **Note** This special image is only supported in 8.3(x); for No Payload Encryption support in 8.4(1) and later, you need to purchase a special hardware version of the ASASM.

Increased contexts for the ASA 5550, 5580, and 5585-X | 8.4(1) | For the ASA 5550 and ASA 5585-X with SSP-10, the maximum contexts was increased from 50 to 100. For the ASA 5580 and 5585-X with SSP-20 and higher, the maximum was increased from 50 to 250.

Increased VLANs for the ASA 5580 and 5585-X | 8.4(1) | For the ASA 5580 and 5585-X, the maximum VLANs was increased from 250 to 1024.

Increased connections for the ASA 5580 and 5585-X | 8.4(1) | We increased the firewall connection limits:

- ASA 5580-20—1,000,000 to 2,000,000.
- ASA 5580-40—2,000,000 to 4,000,000.
- ASA 5585-X with SSP-10: 750,000 to 1,000,000.
- ASA 5585-X with SSP-20: 1,000,000 to 2,000,000.
- ASA 5585-X with SSP-40: 2,000,000 to 4,000,000.
- ASA 5585-X with SSP-60: 2,000,000 to 10,000,000.
### Feature History for Licensing (continued)

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>AnyConnect Premium SSL VPN license changed to AnyConnect Premium license</td>
<td>8.4(1)</td>
<td>The AnyConnect Premium SSL VPN license name was changed to the AnyConnect Premium license. The license information display was changed from “SSL VPN Peers” to “AnyConnect Premium Peers.”</td>
</tr>
<tr>
<td>Increased AnyConnect VPN sessions for the ASA 5580</td>
<td>8.4(1)</td>
<td>The AnyConnect VPN session limit was increased from 5,000 to 10,000.</td>
</tr>
<tr>
<td>Increased Other VPN sessions for the ASA 5580</td>
<td>8.4(1)</td>
<td>The other VPN session limit was increased from 5,000 to 10,000.</td>
</tr>
<tr>
<td>IPsec remote access VPN using IKEv2</td>
<td>8.4(1)</td>
<td>IPsec remote access VPN using IKEv2 was added to the AnyConnect Essentials and AnyConnect Premium licenses. IKEv2 site-to-site sessions were added to the Other VPN license (formerly IPsec VPN). The Other VPN license is included in the Base license.</td>
</tr>
<tr>
<td>No Payload Encryption hardware for export</td>
<td>8.4(1)</td>
<td>For models available with No Payload Encryption (for example, the ASA 5585-X), the ASASM software disables Unified Communications and VPN features, making the ASASM available for export to certain countries.</td>
</tr>
<tr>
<td>Dual SSPs for SSP-20 and SSP-40</td>
<td>8.4(2)</td>
<td>For SSP-40 and SSP-60, you can use two SSPs of the same level in the same chassis. Mixed-level SSPs are not supported (for example, an SSP-40 with an SSP-60 is not supported). Each SSP acts as an independent device, with separate configurations and management. You can use the two SSPs as a failover pair if desired. When using two SSPs in the chassis, VPN is not supported; note, however, that VPN has not been disabled.</td>
</tr>
<tr>
<td>IPS Module license for the ASA 5512-X through ASA 5555-X</td>
<td>8.6(1)</td>
<td>The IPS SSP software module on the ASA 5512-X, ASA 5515-X, ASA 5525-X, ASA 5545-X, and ASA 5555-X requires the IPS module license.</td>
</tr>
</tbody>
</table>
PART 2

Configuring Firewall and Security Context Modes
Configuring the Transparent or Routed Firewall

This chapter describes how to set the firewall mode to routed or transparent, as well as how the firewall works in each firewall mode.

For the ASASM in multiple context mode, you can set the firewall mode independently for each context.

This chapter includes the following sections:

- Configuring the Firewall Mode, page 5-1
- Configuring ARP Inspection for the Transparent Firewall, page 5-9
- Customizing the MAC Address Table for the Transparent Firewall, page 5-13
- Firewall Mode Examples, page 5-17

Configuring the Firewall Mode

This section describes routed and transparent firewall mode, and how to set the mode. This section includes the following topics:

- Information About the Firewall Mode, page 5-1
- Licensing Requirements for the Firewall Mode, page 5-6
- Default Settings, page 5-6
- Guidelines and Limitations, page 5-6
- Setting the Firewall Mode, page 5-8
- Feature History for Firewall Mode, page 5-9

Information About the Firewall Mode

This section describes routed and transparent firewall mode and includes the following topics:

- Information About Routed Firewall Mode, page 5-2
- Information About Transparent Firewall Mode, page 5-2
Information About Routed Firewall Mode

In routed mode, the ASASM is considered to be a router hop in the network. It can use OSPF or RIP (in single context mode). Routed mode supports many interfaces. Each interface is on a different subnet. You can share interfaces between contexts.

The ASASM acts as a router between connected networks, and each interface requires an IP address on a different subnet. In single context mode, the routed firewall supports OSPF, EIGRP, and RIP. Multiple context mode supports static routes only. We recommend using the advanced routing capabilities of the upstream and downstream routers instead of relying on the ASASM for extensive routing needs.

Information About Transparent Firewall Mode

Traditionally, a firewall is a routed hop and acts as a default gateway for hosts that connect to one of its screened subnets. A transparent firewall, on the other hand, is a Layer 2 firewall that acts like a “bump in the wire,” or a “stealth firewall,” and is not seen as a router hop to connected devices.

This section describes transparent firewall mode and includes the following topics:

- Transparent Firewall Network, page 5-2
- Bridge Groups, page 5-2
- Allowing Layer 3 Traffic, page 5-3
- Allowed MAC Addresses, page 5-3
- Passing Traffic Not Allowed in Routed Mode, page 5-3
- BPDU Handling, page 5-4
- MAC Address vs. Route Lookups, page 5-4
- Using the Transparent Firewall in Your Network, page 5-5

Transparent Firewall Network

The ASASM connects the same network between its interfaces. Because the firewall is not a routed hop, you can easily introduce a transparent firewall into an existing network.

Bridge Groups

If you do not want the overhead of security contexts, or want to maximize your use of security contexts, you can group interfaces together in a bridge group, and then configure multiple bridge groups, one for each network. Bridge group traffic is isolated from other bridge groups; traffic is not routed to another bridge group within the ASASM, and traffic must exit the ASASM before it is routed by an external router back to another bridge group in the ASASM. Although the bridging functions are separate for each bridge group, many other functions are shared between all bridge groups. For example, all bridge groups share a syslog server or AAA server configuration. For complete security policy separation, use security contexts with one bridge group in each context.

Note

Each bridge group requires a management IP address. The ASASM uses this IP address as the source address for packets originating from the bridge group. The management IP address must be on the same subnet as the connected network. For another method of management, see the “Allowing Layer 3 Traffic” section on page 5-3.
The ASASM does not support traffic on secondary networks; only traffic on the same network as the management IP address is supported.

Allowing Layer 3 Traffic

- IPv4 and IPv6 traffic is allowed through the transparent firewall automatically from a higher security interface to a lower security interface, without an access list.
- ARPs are allowed through the transparent firewall in both directions without an access list. ARP traffic can be controlled by ARP inspection.
- For Layer 3 traffic travelling from a low to a high security interface, an extended access list is required on the low security interface. See Chapter 14, “Adding an Extended Access List,” or Chapter 17, “Adding an IPv6 Access List,” for more information.

Allowed MAC Addresses

The following destination MAC addresses are allowed through the transparent firewall. Any MAC address not on this list is dropped.

- TRUE broadcast destination MAC address equal to FFFF.FFFF.FFFF
- IPv4 multicast MAC addresses from 0100.5E00.0000 to 0100.5EFE.FFFF
- IPv6 multicast MAC addresses from 3333.0000.0000 to 3333.FFFF.FFFF
- BPDUs multicast address equal to 0100.0CCC.CCCD
- AppleTalk multicast MAC addresses from 0900.0700.0000 to 0900.07FF.FFFF

Passing Traffic Not Allowed in Routed Mode

In routed mode, some types of traffic cannot pass through the ASASM even if you allow it in an access list. The transparent firewall, however, can allow almost any traffic through using either an extended access list (for IP traffic) or an EtherType access list (for non-IP traffic).

Non-IP traffic (for example AppleTalk, IPX, BPDUs, and MPLS) can be configured to go through using an EtherType access list.

<table>
<thead>
<tr>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>The transparent mode ASASM does not pass CDP packets, or any packets that do not have a valid EtherType greater than or equal to 0x600. For example, you cannot pass IS-IS packets. An exception is made for BPDUs, which are supported.</td>
</tr>
</tbody>
</table>

Passing Traffic For Routed-Mode Features

For features that are not directly supported on the transparent firewall, you can allow traffic to pass through so that upstream and downstream routers can support the functionality. For example, by using an extended access list, you can allow DHCP traffic (instead of the unsupported DHCP relay feature) or multicast traffic such as that created by IP/TV. You can also establish routing protocol adjacencies through a transparent firewall; you can allow OSPF, RIP, EIGRP, or BGP traffic through based on an extended access list. Likewise, protocols like HSRP or VRRP can pass through the ASASM.
BPDU Handling

To prevent loops using the Spanning Tree Protocol, BPDUs are passed by default. To block BPDUs, you need to configure an EtherType access list to deny them. If you are using failover, you might want to block BPDUs to prevent the switch port from going into a blocking state when the topology changes. See the “Transparent Firewall Mode Requirements” section on page 49-13 for more information.

MAC Address vs. Route Lookups

When the ASASM runs in transparent mode, the outgoing interface of a packet is determined by performing a MAC address lookup instead of a route lookup.

Route lookups, however, are necessary for the following traffic types:

- Traffic originating on the ASASM—For example, if your syslog server is located on a remote network, you must use a static route so the ASASM can reach that subnet.
- Traffic that is at least one hop away from the ASASM with NAT enabled—The ASASM needs to perform a route lookup; you need to add a static route on the ASASM for the real host address.
- Voice over IP (VoIP) traffic with inspection enabled, and the endpoint is at least one hop away from the ASASM—For example, if you use the transparent firewall between a CCM and an H.323 gateway, and there is a router between the transparent firewall and the H.323 gateway, then you need to add a static route on the ASASM for the H.323 gateway for successful call completion.
- VoIP or DNS traffic with inspection enabled, with NAT enabled, and the embedded address is at least one hop away from the ASASM—To successfully translate the IP address inside VoIP and DNS packets, the ASASM needs to perform a route lookup; you need to add a static route on the ASASM for the real host address that is embedded in the packet.
Using the Transparent Firewall in Your Network

Figure 5-1 shows a typical transparent firewall network where the outside devices are on the same subnet as the inside devices. The inside router and hosts appear to be directly connected to the outside router.

Figure 5-1  Transparent Firewall Network
Figure 5-2 shows two networks connected to the ASASM, which has two bridge groups.

Figure 5-2  Transparent Firewall Network with Two Bridge Groups

Licensing Requirements for the Firewall Mode

The following table shows the licensing requirements for this feature.

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

Default Settings

The default mode is routed mode.

Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

Context Mode Guidelines

- For the ASASM, you can set the firewall mode per context.
- When you change modes, the ASASM clears the running configuration because many commands are not supported for both modes. This action removes any contexts from running. If you then re-add a context that has an existing configuration that was created for the wrong mode, the context
configuration might not work correctly. Be sure to recreate your context configurations for the correct mode before you re-add them, or add new contexts with new paths for the new configurations.

**Transparent Firewall Guidelines**

Follow these guidelines when planning your transparent firewall network:

- In transparent firewall mode, the management interface updates the MAC address table in the same manner as a data interface; therefore you should not connect both a management and a data interface to the same switch unless you configure one of the switch ports as a routed port (by default Cisco Catalyst switches share a MAC address for all VLAN switch ports). Otherwise, if traffic arrives on the management interface from the physically-connected switch, then the ASASM updates the MAC address table to use the management interface to access the switch, instead of the data interface. This action causes a temporary traffic interruption; the ASASM will not re-update the MAC address table for packets from the switch to the data interface for at least 30 seconds for security reasons.

- Each directly-connected network must be on the same subnet.

- Do not specify the bridge group management IP address as the default gateway for connected devices; devices need to specify the router on the other side of the ASASM as the default gateway.

- The default route for the transparent firewall, which is required to provide a return path for management traffic, is only applied to management traffic from one bridge group network. This is because the default route specifies an interface in the bridge group as well as the router IP address on the bridge group network, and you can only define one default route. If you have management traffic from more than one bridge group network, you need to specify a static route that identifies the network from which you expect management traffic.

See the “Guidelines and Limitations” section on page 8-3 for more guidelines.

**IPv6 Guidelines**

Supports IPv6.

**Additional Guidelines and Limitations**

- When you change firewall modes, the ASASM clears the running configuration because many commands are not supported for both modes. The startup configuration remains unchanged. If you reload without saving, then the startup configuration is loaded, and the mode reverts back to the original setting. See the “Setting the Firewall Mode” section on page 5-8 for information about backing up your configuration file.

- If you download a text configuration to the ASASM that changes the mode with the `firewall transparent` command, be sure to put the command at the top of the configuration; the ASASM changes the mode as soon as it reads the command and then continues reading the configuration you downloaded. If the command appears later in the configuration, the ASASM clears all the preceding lines in the configuration. See the “Downloading Software or Configuration Files to Flash Memory” section on page 56-2 for information about downloading text files.

**Unsupported Features in Transparent Mode**

Table 5-1 lists the features are not supported in transparent mode.
Setting the Firewall Mode

This section describes how to change the firewall mode.

Note
We recommend that you set the firewall mode before you perform any other configuration because changing the firewall mode clears the running configuration.

Prerequisites

When you change modes, the ASASM clears the running configuration (see the “Guidelines and Limitations” section on page 5-6 for more information).

- If you already have a populated configuration, be sure to back up your configuration before changing the mode; you can use this backup for reference when creating your new configuration. See the “Backing Up Configuration Files or Other Files” section on page 56-8.
- Use the CLI at the console port to change the mode. If you use any other type of session, including the ASDM Command Line Interface tool or SSH, you will be disconnected when the configuration is cleared, and you will have to reconnect to the ASASM using the console port in any case.
- For the ASASM in multiple context mode, set the mode within the context.
Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>firewall transparent</td>
<td>Sets the firewall mode to transparent. To change the mode to routed, enter the no firewall transparent command.</td>
</tr>
<tr>
<td></td>
<td>Note: You are not prompted to confirm the firewall mode change; the change occurs immediately.</td>
</tr>
</tbody>
</table>

Example:
hostname(config)# firewall transparent

Feature History for Firewall Mode

Table 5-2 lists the release history for each feature change and the platform release in which it was implemented.

Table 5-2 Feature History for Firewall Mode

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transparent firewall mode</td>
<td>7.0(1)</td>
<td>A transparent firewall is a Layer 2 firewall that acts like a “bump in the wire,” or a “stealth firewall,” and is not seen as a router hop to connected devices. We introduced the following commands: firewall transparent, show firewall.</td>
</tr>
<tr>
<td>Transparent firewall bridge groups</td>
<td>8.4(1)</td>
<td>Multiple bridge groups are now allowed in transparent firewall mode. Also, you can now configure up to four interfaces (per bridge group); formerly, you could only configure two interfaces in transparent mode. We introduced the following commands: firewall transparent, show firewall.</td>
</tr>
<tr>
<td>Mixed firewall mode support in multiple context mode for the ASASM only</td>
<td>8.5(1)</td>
<td>You can set the firewall mode independently for each security context in multiple context mode, so some can run in transparent mode while others run in routed mode. We modified the following command: firewall transparent.</td>
</tr>
</tbody>
</table>

Configuring ARP Inspection for the Transparent Firewall

This section describes ARP inspection and how to enable it and includes the following topics:

- Information About ARP Inspection, page 5-10
- Licensing Requirements for ARP Inspection, page 5-10
- Default Settings, page 5-10
Information About ARP Inspection

By default, all ARP packets are allowed through the ASASM. You can control the flow of ARP packets by enabling ARP inspection.

When you enable ARP inspection, the ASASM compares the MAC address, IP address, and source interface in all ARP packets to static entries in the ARP table, and takes the following actions:

- If the IP address, MAC address, and source interface match an ARP entry, the packet is passed through.
- If there is a mismatch between the MAC address, the IP address, or the interface, then the ASASM drops the packet.
- If the ARP packet does not match any entries in the static ARP table, then you can set the ASASM to either forward the packet out all interfaces (flood), or to drop the packet.

**Note**
The dedicated management interface, if present, never floods packets even if this parameter is set to flood.

ARP inspection prevents malicious users from impersonating other hosts or routers (known as ARP spoofing). ARP spoofing can enable a “man-in-the-middle” attack. For example, a host sends an ARP request to the gateway router; the gateway router responds with the gateway router MAC address. The attacker, however, sends another ARP response to the host with the attacker MAC address instead of the router MAC address. The attacker can now intercept all the host traffic before forwarding it on to the router.

ARP inspection ensures that an attacker cannot send an ARP response with the attacker MAC address, so long as the correct MAC address and the associated IP address are in the static ARP table.

Licensing Requirements for ARP Inspection

The following table shows the licensing requirements for this feature.

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

Default Settings

By default, all ARP packets are allowed through the ASASM.

If you enable ARP inspection, the default setting is to flood non-matching packets.
Guidelines and Limitations

Context Mode Guidelines
- Supported in single and multiple context mode.
- In multiple context mode, configure ARP inspection within each context.

Firewall Mode Guidelines
Supported only in transparent firewall mode. Routed mode is not supported.

Configuring ARP Inspection

This section describes how to configure ARP inspection and includes the following topics:
- Task Flow for Configuring ARP Inspection, page 5-11
- Adding a Static ARP Entry, page 5-11
- Enabling ARP Inspection, page 5-12

Task Flow for Configuring ARP Inspection

To configure ARP Inspection, perform the following steps:

Step 1 Add static ARP entries according to the “Adding a Static ARP Entry” section on page 5-11. ARP inspection compares ARP packets with static ARP entries in the ARP table, so static ARP entries are required for this feature.

Step 2 Enable ARP inspection according to the “Enabling ARP Inspection” section on page 5-12.

Adding a Static ARP Entry

ARP inspection compares ARP packets with static ARP entries in the ARP table. Although hosts identify a packet destination by an IP address, the actual delivery of the packet on Ethernet relies on the Ethernet MAC address. When a router or host wants to deliver a packet on a directly connected network, it sends an ARP request asking for the MAC address associated with the IP address, and then delivers the packet to the MAC address according to the ARP response. The host or router keeps an ARP table so it does not have to send ARP requests for every packet it needs to deliver. The ARP table is dynamically updated whenever ARP responses are sent on the network, and if an entry is not used for a period of time, it times out. If an entry is incorrect (for example, the MAC address changes for a given IP address), the entry times out before it can be updated.

Note
The transparent firewall uses dynamic ARP entries in the ARP table for traffic to and from the ASASM, such as management traffic.
**Detailed Steps**

**Command**  
\textbf{arp} \textit{interface\_name ip\_address mac\_address}

**Purpose**  
Adds a static ARP entry.

**Example:**  
hostname(config)# \textbf{arp} outside 10.1.1.1 0009.7cbe.2100

**Examples**

For example, to allow ARP responses from the router at 10.1.1.1 with the MAC address 0009.7cbe.2100 on the outside interface, enter the following command:

hostname(config)# \textbf{arp} outside 10.1.1.1 0009.7cbe.2100

**What to Do Next**

Enable ARP inspection according to the “Enabling ARP Inspection” section on page 5-12.

**Enabling ARP Inspection**

This section describes how to enable ARP inspection.

**Detailed Steps**

**Command**

\textbf{arp\_inspection} \textit{interface\_name enable [flood | no-flood]}

**Purpose**

Enables ARP inspection.

The \textbf{flood} keyword forwards non-matching ARP packets out all interfaces, and \textbf{no-flood} drops non-matching packets.

**Example:**  
hostname(config)# \textbf{arp\_inspection} outside enable no-flood

**Note**

The default setting is to flood non-matching packets. To restrict ARP through the ASASM to only static entries, then set this command to \textbf{no-flood}.

**Examples**

For example, to enable ARP inspection on the outside interface, and to drop all non-matching ARP packets, enter the following command:

hostname(config)# \textbf{arp\_inspection} outside enable no-flood
Monitoring ARP Inspection

To monitor ARP inspection, perform the following task:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show arp-inspection</td>
<td>Shows the current settings for ARP inspection on all interfaces.</td>
</tr>
</tbody>
</table>

Feature History for ARP Inspection

Table 5-2 lists the release history for each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARP inspection</td>
<td>7.0(1)</td>
<td>ARP inspection compares the MAC address, IP address, and source interface in all ARP packets to static entries in the ARP table. We introduced the following commands: <strong>arp</strong>, <strong>arp-inspection</strong>, and <strong>show arp-inspection</strong>.</td>
</tr>
</tbody>
</table>

Customizing the MAC Address Table for the Transparent Firewall

This section describes the MAC address table and includes the following topics:

- Information About the MAC Address Table, page 5-13
- Licensing Requirements for the MAC Address Table, page 5-14
- Default Settings, page 5-14
- Guidelines and Limitations, page 5-14
- Configuring the MAC Address Table, page 5-15
- Monitoring the MAC Address Table, page 5-16
- Feature History for the MAC Address Table, page 5-17

Information About the MAC Address Table

The ASASM learns and builds a MAC address table in a similar way as a normal bridge or switch: when a device sends a packet through the ASASM, the ASASM adds the MAC address to its table. The table associates the MAC address with the source interface so that the ASASM knows to send any packets addressed to the device out the correct interface.
The ASA 5505 includes a built-in switch; the switch MAC address table maintains the MAC address-to-switch port mapping for traffic within each VLAN. This section only discusses the bridge MAC address table, which maintains the MAC address-to-VLAN interface mapping for traffic that passes between VLANs.

Because the ASASM is a firewall, if the destination MAC address of a packet is not in the table, the ASASM does not flood the original packet on all interfaces as a normal bridge does. Instead, it generates the following packets for directly connected devices or for remote devices:

- Packets for directly connected devices—The ASASM generates an ARP request for the destination IP address, so that the ASASM can learn which interface receives the ARP response.
- Packets for remote devices—The ASASM generates a ping to the destination IP address so that the ASASM can learn which interface receives the ping reply.

The original packet is dropped.

### Licensing Requirements for the MAC Address Table

The following table shows the licensing requirements for this feature.

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License</td>
</tr>
</tbody>
</table>

### Default Settings

The default timeout value for dynamic MAC address table entries is 5 minutes.

By default, each interface automatically learns the MAC addresses of entering traffic, and the ASASM adds corresponding entries to the MAC address table.

### Guidelines and Limitations

#### Context Mode Guidelines

- Supported in single and multiple context mode.
- In multiple context mode, configure the MAC address table within each context.

#### Firewall Mode Guidelines

Supported only in transparent firewall mode. Routed mode is not supported.

#### Additional Guidelines

In transparent firewall mode, the management interface updates the MAC address table in the same manner as a data interface; therefore you should not connect both a management and a data interface to the same switch unless you configure one of the switch ports as a routed port (by default Cisco Catalyst switches share a MAC address for all VLAN switch ports). Otherwise, if traffic arrives on the management interface from the physically-connected switch, then the ASASM updates the MAC address.
table to use the management interface to access the switch, instead of the data interface. This action causes a temporary traffic interruption; the ASASM will not re-update the MAC address table for packets from the switch to the data interface for at least 30 seconds for security reasons.

## Configuring the MAC Address Table

This section describes how you can customize the MAC address table and includes the following sections:

- Adding a Static MAC Address, page 5-15
- Setting the MAC Address Timeout, page 5-15
- Disabling MAC Address Learning, page 5-16

### Adding a Static MAC Address

Normally, MAC addresses are added to the MAC address table dynamically as traffic from a particular MAC address enters an interface. You can add static MAC addresses to the MAC address table if desired. One benefit to adding static entries is to guard against MAC spoofing. If a client with the same MAC address as a static entry attempts to send traffic to an interface that does not match the static entry, then the ASASM drops the traffic and generates a system message. When you add a static ARP entry (see the “Adding a Static ARP Entry” section on page 5-11), a static MAC address entry is automatically added to the MAC address table.

To add a static MAC address to the MAC address table, enter the following command:

```
Command Purpose
mac-address-table static interface_name mac_address

Example:
hostname(config)# mac-address-table static inside 0009.7cbe.2100
```

### Setting the MAC Address Timeout

The default timeout value for dynamic MAC address table entries is 5 minutes, but you can change the timeout. To change the timeout, enter the following command:

```
Command Purpose
mac-address-table aging-time timeout_value

Example:
hostname(config)# mac-address-table aging-time 10
```

The timeout (in minutes) is between 5 and 720 (12 hours). 5 minutes is the default.
Disabling MAC Address Learning

By default, each interface automatically learns the MAC addresses of entering traffic, and the ASASM adds corresponding entries to the MAC address table. You can disable MAC address learning if desired, however, unless you statically add MAC addresses to the table, no traffic can pass through the ASASM.

To disable MAC address learning, enter the following command:

```
 hostname(config)# mac-learn interface_name disable
```

Disables MAC address learning.
The `no` form of this command reenables MAC address learning. The `clear configure mac-learn` command reenables MAC address learning on all interfaces.


Monitoring the MAC Address Table

You can view the entire MAC address table (including static and dynamic entries for both interfaces), or you can view the MAC address table for an interface. To view the MAC address table, enter the following command:

```
 hostname# show mac-address-table
```

Shows the MAC address table.

Examples

The following is sample output from the `show mac-address-table` command that shows the entire table:

```
hostname# show mac-address-table
interface    mac address     type     Time Left
outside      0009.7cbe.2100  static   -
inside       0010.7cbe.6101  static   -
inside       0009.7cbe.5101  dynamic  10
```

The following is sample output from the `show mac-address-table` command that shows the table for the inside interface:

```
hostname# show mac-address-table inside
interface    mac address     type     Time Left
inside       0010.7cbe.6101  static   -
inside       0009.7cbe.5101  dynamic  10
```
Feature History for the MAC Address Table

Table 5-2 lists the release history for each feature change and the platform release in which it was implemented.

Table 5-4 Feature History for the MAC Address Table

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAC address table</td>
<td>7.0(1)</td>
<td>Transparent firewall mode uses a MAC address table. We introduced the following commands: mac-address-table static, mac-address-table aging-time, mac-learn disable, and show mac-address-table.</td>
</tr>
</tbody>
</table>

Firewall Mode Examples

This section includes examples of how traffic moves through the ASASM and includes the following topics:

- How Data Moves Through the ASA in Routed Firewall Mode, page 5-17
- How Data Moves Through the Transparent Firewall, page 5-23

How Data Moves Through the ASA in Routed Firewall Mode

This section describes how data moves through the ASASM in routed firewall mode and includes the following topics:

- An Inside User Visits a Web Server, page 5-18
- An Outside User Visits a Web Server on the DMZ, page 5-19
- An Inside User Visits a Web Server on the DMZ, page 5-20
- An Outside User Attempts to Access an Inside Host, page 5-21
- A DMZ User Attempts to Access an Inside Host, page 5-22
An Inside User Visits a Web Server

Figure 5-3 shows an inside user accessing an outside web server.

The following steps describe how data moves through the ASASM (see Figure 5-3):

1. The user on the inside network requests a web page from www.example.com.

2. The ASASM receives the packet and because it is a new session, the ASASM verifies that the packet is allowed according to the terms of the security policy (access lists, filters, AAA).

   For multiple context mode, the ASASM first classifies the packet according to either a unique interface or a unique destination address associated with a context; the destination address is associated by matching an address translation in a context. In this case, the interface would be unique; the www.example.com IP address does not have a current address translation in a context.

3. The ASASM translates the local source address (10.1.2.27) to the global address 209.165.201.10, which is on the outside interface subnet.

   The global address could be on any subnet, but routing is simplified when it is on the outside interface subnet.

4. The ASASM then records that a session is established and forwards the packet from the outside interface.
5. When www.example.com responds to the request, the packet goes through the ASASM, and because the session is already established, the packet bypasses the many lookups associated with a new connection. The ASASM performs NAT by translating the global destination address to the local user address, 10.1.2.27.

6. The ASASM forwards the packet to the inside user.

**An Outside User Visits a Web Server on the DMZ**

Figure 5-4 shows an outside user accessing the DMZ web server.

![Figure 5-4](image)

The following steps describe how data moves through the ASASM (see Figure 5-4):

1. A user on the outside network requests a web page from the DMZ web server using the global destination address of 209.165.201.3, which is on the outside interface subnet.

2. The ASASM untranslated the destination address to the local address 10.1.1.3.

3. The ASASM receives the packet and because it is a new session, the ASASM verifies that the packet is allowed according to the terms of the security policy (access lists, filters, AAA).

   For multiple context mode, the ASASM first classifies the packet according to either a unique interface or a unique destination address associated with a context; the destination address is associated by matching an address translation in a context. In this case, the classifier “knows” that the DMZ web server address belongs to a certain context because of the server address translation.
4. The ASASM then adds a session entry to the fast path and forwards the packet from the DMZ interface.

5. When the DMZ web server responds to the request, the packet goes through the ASASM and because the session is already established, the packet bypasses the many lookups associated with a new connection. The ASASM performs NAT by translating the local source address to 209.165.201.3.

6. The ASASM forwards the packet to the outside user.

An Inside User Visits a Web Server on the DMZ

Figure 5-5 shows an inside user accessing the DMZ web server.

The following steps describe how data moves through the ASASM (see Figure 5-5):

1. A user on the inside network requests a web page from the DMZ web server using the destination address of 10.1.1.3.

2. The ASASM receives the packet and because it is a new session, the ASASM verifies that the packet is allowed according to the terms of the security policy (access lists, filters, AAA).

   For multiple context mode, the ASASM first classifies the packet according to either a unique interface or a unique destination address associated with a context; the destination address is associated by matching an address translation in a context. In this case, the interface is unique; the web server IP address does not have a current address translation.
3. The ASASM then records that a session is established and forwards the packet out of the DMZ interface.

4. When the DMZ web server responds to the request, the packet goes through the fast path, which lets the packet bypass the many lookups associated with a new connection.

5. The ASASM forwards the packet to the inside user.

**An Outside User Attempts to Access an Inside Host**

Figure 5-6 shows an outside user attempting to access the inside network.

The following steps describe how data moves through the ASASM (see Figure 5-6):

1. A user on the outside network attempts to reach an inside host (assuming the host has a routable IP address).

   If the inside network uses private addresses, no outside user can reach the inside network without NAT. The outside user might attempt to reach an inside user by using an existing NAT session.

2. The ASASM receives the packet and because it is a new session, the ASASM verifies if the packet is allowed according to the security policy (access lists, filters, AAA).

3. The packet is denied, and the ASASM drops the packet and logs the connection attempt.

   If the outside user is attempting to attack the inside network, the ASASM employs many technologies to determine if a packet is valid for an already established session.
A DMZ User Attempts to Access an Inside Host

Figure 5-7 shows a user in the DMZ attempting to access the inside network.

The following steps describe how data moves through the ASASM (see Figure 5-7):

1. A user on the DMZ network attempts to reach an inside host. Because the DMZ does not have to route the traffic on the Internet, the private addressing scheme does not prevent routing.

2. The ASASM receives the packet and because it is a new session, the ASASM verifies if the packet is allowed according to the security policy (access lists, filters, AAA).

   The packet is denied, and the ASASM drops the packet and logs the connection attempt.
How Data Moves Through the Transparent Firewall

Figure 5-8 shows a typical transparent firewall implementation with an inside network that contains a public web server. The ASASM has an access list so that the inside users can access Internet resources. Another access list lets the outside users access only the web server on the inside network.

This section describes how data moves through the ASASM and includes the following topics:

- An Inside User Visits a Web Server, page 5-24
- An Inside User Visits a Web Server Using NAT, page 5-25
- An Outside User Visits a Web Server on the Inside Network, page 5-26
- An Outside User Attempts to Access an Inside Host, page 5-27
An Inside User Visits a Web Server

Figure 5-9 shows an inside user accessing an outside web server.

The following steps describe how data moves through the ASASM (see Figure 5-9):

1. The user on the inside network requests a web page from www.example.com.
2. The ASASM receives the packet and adds the source MAC address to the MAC address table, if required. Because it is a new session, it verifies that the packet is allowed according to the terms of the security policy (access lists, filters, AAA).
   For multiple context mode, the ASASM first classifies the packet according to a unique interface.
3. The ASASM records that a session is established.
4. If the destination MAC address is in its table, the ASASM forwards the packet out of the outside interface. The destination MAC address is that of the upstream router, 209.165.201.2.
   If the destination MAC address is not in the ASASM table, the ASASM attempts to discover the MAC address by sending an ARP request or a ping. The first packet is dropped.
5. The web server responds to the request; because the session is already established, the packet bypasses the many lookups associated with a new connection.
6. The ASASM forwards the packet to the inside user.
An Inside User Visits a Web Server Using NAT

Figure 5-10 shows an inside user accessing an outside web server.

Figure 5-10  Inside to Outside with NAT

The following steps describe how data moves through the ASASM (see Figure 5-10):

1. The user on the inside network requests a web page from www.example.com.
2. The ASASM receives the packet and adds the source MAC address to the MAC address table, if required. Because it is a new session, it verifies that the packet is allowed according to the terms of the security policy (access lists, filters, AAA).
   
   For multiple context mode, the ASASM first classifies the packet according to a unique interface.
3. The ASASM translates the real address (10.1.2.27) to the mapped address 209.165.201.10.
   
   Because the mapped address is not on the same network as the outside interface, then be sure the upstream router has a static route to the mapped network that points to the ASASM.
4. The ASASM then records that a session is established and forwards the packet from the outside interface.
5. If the destination MAC address is in its table, the ASASM forwards the packet out of the outside interface. The destination MAC address is that of the upstream router, 10.1.2.1.
   
   If the destination MAC address is not in the ASASM table, the ASASM attempts to discover the MAC address by sending an ARP request and a ping. The first packet is dropped.
6. The web server responds to the request; because the session is already established, the packet bypasses the many lookups associated with a new connection.
7. The ASASM performs NAT by translating the mapped address to the real address, 10.1.2.27.
An Outside User Visits a Web Server on the Inside Network

Figure 5-11 shows an outside user accessing the inside web server.

The following steps describe how data moves through the ASASM (see Figure 5-11):

1. A user on the outside network requests a web page from the inside web server.

2. The ASASM receives the packet and adds the source MAC address to the MAC address table, if required. Because it is a new session, it verifies that the packet is allowed according to the terms of the security policy (access lists, filters, AAA).

   For multiple context mode, the ASASM first classifies the packet according to a unique interface.

3. The ASASM records that a session is established.

4. If the destination MAC address is in its table, the ASASM forwards the packet out of the inside interface. The destination MAC address is that of the downstream router, 209.165.201.1.

   If the destination MAC address is not in the ASASM table, the ASASM attempts to discover the MAC address by sending an ARP request and a ping. The first packet is dropped.

5. The web server responds to the request; because the session is already established, the packet bypasses the many lookups associated with a new connection.

6. The ASASM forwards the packet to the outside user.
An Outside User Attempts to Access an Inside Host

Figure 5-12 shows an outside user attempting to access a host on the inside network.

Figure 5-12 Outside to Inside

The following steps describe how data moves through the ASASM (see Figure 5-12):

1. A user on the outside network attempts to reach an inside host.

2. The ASASM receives the packet and adds the source MAC address to the MAC address table, if required. Because it is a new session, it verifies if the packet is allowed according to the terms of the security policy (access lists, filters, AAA).

   For multiple context mode, the ASASM first classifies the packet according to a unique interface.

3. The packet is denied because there is no access list permitting the outside host, and the ASASM drops the packet.

4. If the outside user is attempting to attack the inside network, the ASASM employs many technologies to determine if a packet is valid for an already established session.
Configuring Multiple Context Mode

This chapter describes how to configure multiple security contexts on the ASASM and includes the following sections:

- Information About Security Contexts, page 6-1
- Licensing Requirements for Multiple Context Mode, page 6-12
- Guidelines and Limitations, page 6-13
- Default Settings, page 6-13
- Configuring Multiple Contexts, page 6-14
- Changing Between Contexts and the System Execution Space, page 6-21
- Managing Security Contexts, page 6-22
- Monitoring Security Contexts, page 6-25
- Configuration Examples for Multiple Context Mode, page 6-36
- Feature History for Multiple Context Mode, page 6-37

Information About Security Contexts

You can partition a single ASASM into multiple virtual devices, known as security contexts. Each context is an independent device, with its own security policy, interfaces, and administrators. Multiple contexts are similar to having multiple standalone devices. Many features are supported in multiple context mode, including routing tables, firewall features, IPS, and management. Some features are not supported, including VPN and dynamic routing protocols.

Note

When the ASASM is configured for security contexts (for example, for Active/Active Stateful Failover), IPsec or SSL VPN cannot be enabled. Therefore, these features are unavailable.

This section provides an overview of security contexts and includes the following topics:

- Common Uses for Security Contexts, page 6-2
- Context Configuration Files, page 6-2
- How the ASA Classifies Packets, page 6-3
- Cascading Security Contexts, page 6-6
- Management Access to Security Contexts, page 6-7
Common Uses for Security Contexts

You might want to use multiple security contexts in the following situations:

- You are a service provider and want to sell security services to many customers. By enabling multiple security contexts on the ASASM, you can implement a cost-effective, space-saving solution that keeps all customer traffic separate and secure, and also eases configuration.
- You are a large enterprise or a college campus and want to keep departments completely separate.
- You are an enterprise that wants to provide distinct security policies to different departments.
- You have any network that requires more than one ASASM.

Context Configuration Files

This section describes how the ASASM implements multiple context mode configurations and includes the following sections:

- Context Configurations, page 6-2
- System Configuration, page 6-2
- Admin Context Configuration, page 6-2

Context Configurations

The ASASM includes a configuration for each context that identifies the security policy, interfaces, and almost all the options you can configure on a standalone device. You can store context configurations on the internal flash memory or the external flash memory card, or you can download them from a TFTP, FTP, or HTTP(S) server.

System Configuration

The system administrator adds and manages contexts by configuring each context configuration location, allocated interfaces, and other context operating parameters in the system configuration, which, like a single mode configuration, is the startup configuration. The system configuration identifies basic settings for the ASASM. The system configuration does not include any network interfaces or network settings for itself; rather, when the system needs to access network resources (such as downloading the contexts from the server), it uses one of the contexts that is designated as the admin context. The system configuration does include a specialized failover interface for failover traffic only.

Admin Context Configuration

The admin context is just like any other context, except that when a user logs in to the admin context, then that user has system administrator rights and can access the system and all other contexts. The admin context is not restricted in any way, and can be used as a regular context. However, because
Information About Security Contexts

logging into the admin context grants you administrator privileges over all contexts, you might need to restrict access to the admin context to appropriate users. The admin context must reside on flash memory, and not remotely.

If your system is already in multiple context mode, or if you convert from single mode, the admin context is created automatically as a file on the internal flash memory called admin.cfg. This context is named “admin.” If you do not want to use admin.cfg as the admin context, you can change the admin context.

How the ASA Classifies Packets

Each packet that enters the ASASM must be classified, so that the ASASM can determine to which context to send a packet. This section includes the following topics:

- Valid Classifier Criteria, page 6-3
- Classification Examples, page 6-4

Note: If the destination MAC address is a multicast or broadcast MAC address, the packet is duplicated and delivered to each context.

Valid Classifier Criteria

This section describes the criteria used by the classifier and includes the following topics:

- Unique Interfaces, page 6-3
- Unique MAC Addresses, page 6-3
- NAT Configuration, page 6-4

Note: For management traffic destined for an interface, the interface IP address is used for classification.

The routing table is not used for packet classification.

Unique Interfaces

If only one context is associated with the ingress interface, the ASASM classifies the packet into that context. In transparent firewall mode, unique interfaces for contexts are required, so this method is used to classify packets at all times.

Unique MAC Addresses

If multiple contexts share an interface, then the classifier uses the interface MAC address, or for the ASASM, the backplane MAC address. The ASASM lets you assign a different MAC address in each context to the same shared interface. By default, shared interfaces do not have unique MAC addresses; the interface uses the burned-in MAC address in every context. An upstream router cannot route directly to a context without unique MAC addresses. You can set the MAC addresses manually when you configure each interface (see the “Configuring the MAC Address and MTU” section on page 7-6), or you can automatically generate MAC addresses (see the “Automatically Assigning MAC Addresses to Context Interfaces” section on page 6-21).
NAT Configuration

If you do not use unique MAC addresses, then the mapped addresses in your NAT configuration are used to classify packets. We recommend using MAC addresses instead of NAT, so that traffic classification can occur regardless of the completeness of the NAT configuration.

Classification Examples

Figure 6-1 shows multiple contexts sharing an outside interface. The classifier assigns the packet to Context B because Context B includes the MAC address to which the router sends the packet.

Figure 6-1  Packet Classification with a Shared Interface using MAC Addresses
Note that all new incoming traffic must be classified, even from inside networks. Figure 6-2 shows a host on the Context B inside network accessing the Internet. The classifier assigns the packet to Context B because the ingress interface is Gigabit Ethernet 0/1.3, which is assigned to Context B.

Figure 6-2  Incoming Traffic from Inside Networks
For transparent firewalls, you must use unique interfaces. Figure 6-3 shows a host on the Context B inside network accessing the Internet. The classifier assigns the packet to Context B because the ingress interface is Gigabit Ethernet 1/0.3, which is assigned to Context B.

**Figure 6-3  Transparent Firewall Contexts**

**Cascading Security Contexts**

Placing a context directly in front of another context is called cascading contexts; the outside interface of one context is the same interface as the inside interface of another context. You might want to cascade contexts if you want to simplify the configuration of some contexts by configuring shared parameters in the top context.

---

**Note**
Cascading contexts requires that you configure unique MAC addresses for each context interface. Because of the limitations of classifying packets on shared interfaces without MAC addresses, we do not recommend using cascading contexts without unique MAC addresses.
Figure 6-4 shows a gateway context with two contexts behind the gateway.

**Figure 6-4 Cascading Contexts**

![Diagram of cascading contexts]

**Management Access to Security Contexts**

The ASASM provides system administrator access in multiple context mode as well as access for individual context administrators. The following sections describe logging in as a system administrator or as a context administrator:

- System Administrator Access, page 6-7
- Context Administrator Access, page 6-8

**System Administrator Access**

You can access the ASASM as a system administrator in two ways:

- Access the ASASM console.
  
  From the console, you access the system execution space, which means that any commands you enter affect only the system configuration or the running of the system (for run-time commands).

- Access the admin context using Telnet, SSH, or ASDM.
  
  See Chapter 34, “Configuring Management Access,” to enable Telnet, SSH, and SDM access.

As the system administrator, you can access all contexts.

When you change to a context from admin or the system, your username changes to the default “enable_15” username. If you configured command authorization in that context, you need to either configure authorization privileges for the “enable_15” user, or you can log in as a different name for which you provide sufficient privileges in the command authorization configuration for the context. To
log in with a username, enter the `login` command. For example, you log in to the admin context with the username “admin.” The admin context does not have any command authorization configuration, but all other contexts include command authorization. For convenience, each context configuration includes a user “admin” with maximum privileges. When you change from the admin context to context A, your username is altered, so you must log in again as “admin” by entering the `login` command. When you change to context B, you must again enter the `login` command to log in as “admin.”

The system execution space does not support any AAA commands, but you can configure its own enable password, as well as usernames in the local database to provide individual logins.

**Context Administrator Access**

You can access a context using Telnet, SSH, or ASDM. If you log in to a non-admin context, you can only access the configuration for that context. You can provide individual logins to the context. See Chapter 34, “Configuring Management Access,” to enable Telnet, SSH, and SDM access and to configure management authentication.

**Information About Resource Management**

By default, all security contexts have unlimited access to the resources of the ASASM, except where maximum limits per context are enforced. However, if you find that one or more contexts use too many resources, and they cause other contexts to be denied connections, for example, then you can configure resource management to limit the use of resources per context.

The ASASM manages resources by assigning contexts to resource classes. Each context uses the resource limits set by the class.

This section includes the following topics:

- Resource Limits, page 6-8
- Default Class, page 6-9
- Class Members, page 6-10

**Resource Limits**

When you create a class, the ASASM does not set aside a portion of the resources for each context assigned to the class; rather, the ASASM sets the maximum limit for a context. If you oversubscribe resources, or allow some resources to be unlimited, a few contexts can “use up” those resources, potentially affecting service to other contexts.

You can set the limit for individual resources, as a percentage (if there is a hard system limit) or as an absolute value.

You can oversubscribe the ASASM by assigning more than 100 percent of a resource across all contexts. For example, you can set the Bronze class to limit connections to 20 percent per context, and then assign 10 contexts to the class for a total of 200 percent. If contexts concurrently use more than the system limit, then each context gets less than the 20 percent you intended. (See Figure 6-5.)
If you assign an absolute value to a resource across all contexts that exceeds the practical limit of the ASASM, then the performance of the ASASM might be impaired.

The ASASM lets you assign unlimited access to one or more resources in a class, instead of a percentage or absolute number. When a resource is unlimited, contexts can use as much of the resource as the system has available or that is practically available. For example, Context A, B, and C are in the Silver Class, which limits each class member to 1 percent of the connections, for a total of 3 percent; but the three contexts are currently only using 2 percent combined. Gold Class has unlimited access to connections. The contexts in the Gold Class can use more than the 97 percent of "unassigned" connections; they can also use the 1 percent of connections not currently in use by Context A, B, and C, even if that means that Context A, B, and C are unable to reach their 3 percent combined limit. (See Figure 6-6.) Setting unlimited access is similar to oversubscribing the ASASM, except that you have less control over how much you oversubscribe the system.

**Figure 6-5 Resource Oversubscription**

If you assign an absolute value to a resource across all contexts that exceeds the practical limit of the ASASM, then the performance of the ASASM might be impaired.

The ASASM lets you assign unlimited access to one or more resources in a class, instead of a percentage or absolute number. When a resource is unlimited, contexts can use as much of the resource as the system has available or that is practically available. For example, Context A, B, and C are in the Silver Class, which limits each class member to 1 percent of the connections, for a total of 3 percent; but the three contexts are currently only using 2 percent combined. Gold Class has unlimited access to connections. The contexts in the Gold Class can use more than the 97 percent of "unassigned" connections; they can also use the 1 percent of connections not currently in use by Context A, B, and C, even if that means that Context A, B, and C are unable to reach their 3 percent combined limit. (See Figure 6-6.) Setting unlimited access is similar to oversubscribing the ASASM, except that you have less control over how much you oversubscribe the system.

**Figure 6-6 Unlimited Resources**

If you assign an absolute value to a resource across all contexts that exceeds the practical limit of the ASASM, then the performance of the ASASM might be impaired.

The ASASM lets you assign unlimited access to one or more resources in a class, instead of a percentage or absolute number. When a resource is unlimited, contexts can use as much of the resource as the system has available or that is practically available. For example, Context A, B, and C are in the Silver Class, which limits each class member to 1 percent of the connections, for a total of 3 percent; but the three contexts are currently only using 2 percent combined. Gold Class has unlimited access to connections. The contexts in the Gold Class can use more than the 97 percent of "unassigned" connections; they can also use the 1 percent of connections not currently in use by Context A, B, and C, even if that means that Context A, B, and C are unable to reach their 3 percent combined limit. (See Figure 6-6.) Setting unlimited access is similar to oversubscribing the ASASM, except that you have less control over how much you oversubscribe the system.

**Default Class**

All contexts belong to the default class if they are not assigned to another class; you do not have to actively assign a context to the default class.
If a context belongs to a class other than the default class, those class settings always override the default class settings. However, if the other class has any settings that are not defined, then the member context uses the default class for those limits. For example, if you create a class with a 2 percent limit for all concurrent connections, but no other limits, then all other limits are inherited from the default class. Conversely, if you create a class with a limit for all resources, the class uses no settings from the default class.

By default, the default class provides unlimited access to resources for all contexts, except for the following limits, which are by default set to the maximum allowed per context:

- Telnet sessions—5 sessions.
- SSH sessions—5 sessions.
- IPsec sessions—5 sessions.
- MAC addresses—65,535 entries.

Figure 6-7 shows the relationship between the default class and other classes. Contexts A and C belong to classes with some limits set; other limits are inherited from the default class. Context B inherits no limits from default because all limits are set in its class, the Gold class. Context D was not assigned to a class, and is by default a member of the default class.

Class Members

To use the settings of a class, assign the context to the class when you define the context. All contexts belong to the default class if they are not assigned to another class; you do not have to actively assign a context to default. You can only assign a context to one resource class. The exception to this rule is that limits that are undefined in the member class are inherited from the default class; so in effect, a context could be a member of default plus another class.
Information About Security Contexts

Information About MAC Addresses

To allow contexts to share interfaces, you should assign unique MAC addresses to each shared context interface.

The MAC address is used to classify packets within a context. If you share an interface, but do not have unique MAC addresses for the interface in each context, then other classification methods are attempted that might not provide full coverage. See the “How the ASA Classifies Packets” section on page 6-3 for information about classifying packets.

In the rare circumstance that the generated MAC address conflicts with another private MAC address in your network, you can manually set the MAC address for the interface within the context. See the “Configuring the MAC Address and MTU” section on page 7-6 to manually set the MAC address.

This section includes the following topics:

- Default MAC Address, page 6-11
- Interaction with Manual MAC Addresses, page 6-11
- Failover MAC Addresses, page 6-12
- MAC Address Format, page 6-12

Default MAC Address

If you disable MAC address generation, all VLAN interfaces use the same MAC address, derived from the backplane MAC address.

See the following sections for your release for additional information about automatic MAC address generation. See also the “MAC Address Format” section on page 6-12.

8.5(1.7) and Later

Automatic MAC address generation is enabled—Uses an autogenerated prefix. The ASASM autogenerates the prefix based on the last two bytes of the backplane MAC address. You cannot use the legacy auto-generation method (without a prefix).

Note

To maintain hitless upgrade for failover pairs, the ASASM does not convert an existing auto-generation configuration upon a reload if failover is enabled. However, we strongly recommend that you manually change to the prefix method of generation when using failover. Without the prefix method, ASASMs installed in different slot numbers experience a MAC address change upon failover, and can experience traffic interruption. After upgrading, to use the prefix method of MAC address generation, reenable MAC address autogeneration to use a prefix.

Earlier Releases

Automatic MAC address generation is enabled—Uses the legacy auto-generation method (without a prefix).

Interaction with Manual MAC Addresses

If you manually assign a MAC address and also enable auto-generation, then the manually assigned MAC address is used. If you later remove the manual MAC address, the auto-generated address is used.

Because auto-generated addresses (when using a prefix) start with A2, you cannot start manual MAC addresses with A2 if you also want to use auto-generation.
Failover MAC Addresses

For use with failover, the ASASM generates both an active and standby MAC address for each interface. If the active unit fails over and the standby unit becomes active, the new active unit starts using the active MAC addresses to minimize network disruption. See the “MAC Address Format” section for more information.

MAC Address Format

The MAC address format without a prefix is a legacy version not supported on newer ASA versions.

MAC Address Format Using a Prefix

The ASASM generates the MAC address using the following format:

A2\text{xx.\text{yy}.\text{zz}.\text{zzz}}

Where \text{xx.\text{yy}} is a user-defined prefix or an autogenerated prefix based on the last two bytes of the backplane MAC address, and \text{zz.\text{zzz}} is an internal counter generated by the ASASM. For the standby MAC address, the address is identical except that the internal counter is increased by 1.

For an example of how the prefix is used, if you set a prefix of 77, then the ASASM converts 77 into the hexadecimal value \text{004D (\text{yyxx})}. When used in the MAC address, the prefix is reversed (\text{xxyy}) to match the ASASM native form:

A2\text{4D.00.\text{zz}.\text{zzz}}

For a prefix of 1009 (\text{03F1}), the MAC address is:

A2\text{F1.03.\text{zz}.\text{zzz}}

MAC Address Format Without a Prefix (Legacy Method; Not Available in 8.5(1.7) and Later)

Without a prefix, the MAC address is generated using the following format:

- Active unit MAC address: 12_{\text{slot.port_subid.contextid}}.
- Standby unit MAC address: 02_{\text{slot.port_subid.contextid}}.

The \text{port} is the interface port. The \text{subid} is an internal ID for the subinterface, which is not viewable. The \text{contextid} is an internal ID for the context, viewable with the \text{show context detail} command. For example, the interface GigabitEthernet 0/1.200 in the context with the ID 1 has the following generated MAC addresses, where the internal ID for subinterface 200 is 31:

- Active: 1200.0131.0001
- Standby: 0200.0131.0001

This MAC address generation method does not allow for persistent MAC addresses across reboots, does not allow for multiple ASASMs on the same network segment (because unique MAC addresses are not guaranteed), and does not prevent overlapping MAC addresses with manually assigned MAC addresses. We recommend using a prefix with the MAC address generation to avoid these issues.

Licensing Requirements for Multiple Context Mode
### Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

**Firewall Mode Guidelines**
Supported in routed and transparent firewall mode.

**Failover Guidelines**
Active/Active mode failover is only supported in multiple context mode.

**IPv6 Guidelines**
Supports IPv6.

**Model Guidelines**
Does not support the ASA 5505.

**Unsupported Features**
Multiple context mode does not support the following features:
- Dynamic routing protocols
  - Security contexts support only static routes. You cannot enable OSPF, RIP, or EIGRP in multiple context mode.
- VPN
- Multicast routing
- Threat Detection
- Phone Proxy
- QoS
- Unified Communications

**Additional Guidelines**
The context mode (single or multiple) is not stored in the configuration file, even though it does endure reboots. If you need to copy your configuration to another device, set the mode on the new device to match.

### Default Settings

By default, the ASASM is in single context mode.

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASASM</td>
<td>Base License: 2 contexts.</td>
</tr>
<tr>
<td></td>
<td>Optional licenses: 5, 10, 20, 50, 100, or 250 contexts.</td>
</tr>
</tbody>
</table>
Configuring Multiple Contexts

This section describes how to configure multiple context mode, and includes the following topics:

- Task Flow for Configuring Multiple Context Mode, page 6-14
- Enabling or Disabling Multiple Context Mode, page 6-14
- Configuring a Class for Resource Management, page 6-15
- Configuring a Security Context, page 6-17
- Automatically Assigning MAC Addresses to Context Interfaces, page 6-21

Task Flow for Configuring Multiple Context Mode

To configure multiple context mode, perform the following steps:

**Step 1**  Enable multiple context mode. See the “Enabling or Disabling Multiple Context Mode” section on page 6-14.

**Step 2**  (Optional) Configure classes for resource management. See the “Configuring a Class for Resource Management” section on page 6-15.

**Step 3**  Configure security contexts. See the “Configuring a Security Context” section on page 6-17.

**Step 4**  (Optional) Automatically assign MAC addresses to context interfaces. See the “Automatically Assigning MAC Addresses to Context Interfaces” section on page 6-21.

**Step 5**  Complete interface configuration in the context. See Chapter 7, “Configuring Interfaces (Routed Mode),” or Chapter 8, “Configuring Interfaces (Transparent Mode).”

Enabling or Disabling Multiple Context Mode

Your ASASM might already be configured for multiple security contexts depending on how you ordered it from Cisco. If you are upgrading, however, you might need to convert from single mode to multiple mode by following the procedures in this section.

This section includes the following topics:

- Enabling Multiple Context Mode, page 6-14
- Restoring Single Context Mode, page 6-15

Enabling Multiple Context Mode

When you convert from single mode to multiple mode, the ASASM converts the running configuration into two files: a new startup configuration that comprises the system configuration, and admin.cfg that comprises the admin context (in the root directory of the internal flash memory). The original running configuration is saved as old_running.cfg (in the root directory of the internal flash memory). The original startup configuration is not saved. The ASASM automatically adds an entry for the admin context to the system configuration with the name “admin.”
Prerequisites

- When you convert from single mode to multiple mode, the ASASM converts the running configuration into two files. The original startup configuration is not saved, so if it differs from the running configuration, you should back it up before proceeding.
- The context mode (single or multiple) is not stored in the configuration file, even though it does endure reboots. If you need to copy your configuration to another device, set the mode on the new device to match.

Detailed Steps

### Restoring Single Context Mode

To copy the old running configuration to the startup configuration and to change the mode to single mode, perform the following steps.

**Prerequisites**

Perform this procedure in the system execution space.

**Detailed Steps**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>copy flash:old_running.cfg startup-config</td>
<td>Copies the backup version of your original running configuration to the current startup configuration.</td>
</tr>
</tbody>
</table>

**Example:**

```
hostname(config)# copy flash:old_running.cfg startup-config
```

**Step 2**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>mode single</td>
<td>Sets the mode to single mode. You are prompted to reboot the ASASM.</td>
</tr>
</tbody>
</table>

**Example:**

```
hostname(config)# mode single
```

### Configuring a Class for Resource Management

To configure a class in the system configuration, perform the following steps. You can change the value of a particular resource limit by reentering the command with a new value.
Prerequisites

Perform this procedure in the system execution space.

Guidelines

Table 6-1 lists the resource types and the limits. See also the `show resource types` command.

<table>
<thead>
<tr>
<th>Resource Name</th>
<th>Rate or Concurrent</th>
<th>Minimum and Maximum Number per Context</th>
<th>System Limit¹</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>mac-addresses</td>
<td>Concurrent</td>
<td>N/A</td>
<td>65,535</td>
<td>For transparent firewall mode, the number of MAC addresses allowed in the MAC address table.</td>
</tr>
<tr>
<td>conns</td>
<td>Concurrent or Rate</td>
<td>N/A</td>
<td>Concurrent connections: See the “Supported Feature Licenses” section on page 4-1 for the connection limit for your platform. Rate: N/A</td>
<td>TCP or UDP connections between any two hosts, including connections between one host and multiple other hosts.</td>
</tr>
<tr>
<td>inspects</td>
<td>Rate</td>
<td>N/A</td>
<td>N/A</td>
<td>Application inspections.</td>
</tr>
<tr>
<td>hosts</td>
<td>Concurrent</td>
<td>N/A</td>
<td>N/A</td>
<td>Hosts that can connect through the ASASM.</td>
</tr>
<tr>
<td>asdm</td>
<td>Concurrent</td>
<td>1 minimum 5 maximum</td>
<td>200</td>
<td>ASDM management sessions. Note ASDM sessions use two HTTPS connections: one for monitoring that is always present, and one for making configuration changes that is present only when you make changes. For example, the system limit of 32 ASDM sessions represents a limit of 64 HTTPS sessions.</td>
</tr>
<tr>
<td>ssh</td>
<td>Concurrent</td>
<td>1 minimum 5 maximum</td>
<td>100</td>
<td>SSH sessions.</td>
</tr>
<tr>
<td>syslog</td>
<td>Rate</td>
<td>N/A</td>
<td>N/A</td>
<td>Syslog messages.</td>
</tr>
<tr>
<td>telnet</td>
<td>Concurrent</td>
<td>1 minimum 5 maximum</td>
<td>100</td>
<td>Telnet sessions.</td>
</tr>
<tr>
<td>xlate</td>
<td>Concurrent</td>
<td>N/A</td>
<td>N/A</td>
<td>Address translations.</td>
</tr>
</tbody>
</table>

¹ If this column value is N/A, then you cannot set a percentage of the resource because there is no hard system limit for the resource.
Chapter 6  Configuring Multiple Context Mode

Configuring Multiple Contexts

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong>&lt;br&gt;<code>class name</code>&lt;br&gt;&lt;br&gt;<strong>Example:</strong>&lt;br&gt;hostname(config)# threat-detection scanning-threat shun except ip-address 10.1.1.0 255.255.255.0</td>
<td>Specifies the class name and enters the class configuration mode. The <code>name</code> is a string up to 20 characters long. To set the limits for the default class, enter <code>default</code> for the name.</td>
</tr>
<tr>
<td><strong>Step 2</strong>&lt;br&gt;Do one or more of the following:&lt;br&gt;&lt;br&gt;<code>limit-resource all 0</code>&lt;br&gt;&lt;br&gt;<strong>Example:</strong>&lt;br&gt;hostname(config)# limit-resource all 0</td>
<td>Sets all resource limits (shown in Table 6-1) to be unlimited. For example, you might want to create a class that includes the admin context that has no limitations. The default class has all resources set to unlimited by default.</td>
</tr>
<tr>
<td><code>limit-resource [rate] resource_name number [%]</code>&lt;br&gt;&lt;br&gt;<strong>Example:</strong>&lt;br&gt;hostname(config)# limit-resource rate inspects 10</td>
<td>Sets a particular resource limit. For this particular resource, the limit overrides the limit set for all. Enter the <code>rate</code> argument to set the rate per second for certain resources. For resources that do not have a system limit, you cannot set the percentage (%) between 1 and 100; you can only set an absolute value. See Table 6-1 for resources for which you can set the rate per second and which do not have a system limit.</td>
</tr>
</tbody>
</table>

Examples

For example, to set the default class limit for conns to 10 percent instead of unlimited, enter the following commands:

```bash
hostname(config)# class default
hostname(config-class)# limit-resource conns 10%
```

All other resources remain at unlimited.

To add a class called gold, enter the following commands:

```bash
hostname(config)# class gold
hostname(config-class)# limit-resource mac-addresses 10000
hostname(config-class)# limit-resource conns 15%
hostname(config-class)# limit-resource rate conns 1000
hostname(config-class)# limit-resource rate inspects 500
hostname(config-class)# limit-resource hosts 9000
hostname(config-class)# limit-resource asdm 5
hostname(config-class)# limit-resource ssh 5
hostname(config-class)# limit-resource rate syslogs 5000
hostname(config-class)# limit-resource telnet 5
hostname(config-class)# limit-resource xlates 36000
```

Configuring a Security Context

The security context definition in the system configuration identifies the context name, configuration file URL, and interfaces that a context can use.
Prerequisites

- Perform this procedure in the system execution space.
- For the ASASM, assign VLANs to the ASASM on the switch according to Chapter 2, “Configuring the Switch for Use with the ASA Services Module.”
- If you do not have an admin context (for example, if you clear the configuration) then you must first specify the admin context name by entering the following command:

  `hostname(config)# admin-context name`

  Although this context name does not exist yet in your configuration, you can subsequently enter the `context name` command to match the specified name to continue the admin context configuration.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td><code>context name</code></td>
<td>Adds or modifies a context. The <code>name</code> is a string up to 32 characters long. This name is case sensitive, so you can have two contexts named “customerA” and “CustomerA,” for example. You can use letters, digits, or hyphens, but you cannot start or end the name with a hyphen. “System” or “Null” (in upper or lower case letters) are reserved names, and cannot be used.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>hostname(config)# context administrator</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td><code>(Optional)</code></td>
<td></td>
</tr>
<tr>
<td><code>description text</code></td>
<td>Adds a description for this context.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>hostname(config)# description Administrator Context</code></td>
<td></td>
</tr>
</tbody>
</table>
### Chapter 6  Configuring Multiple Context Mode

### Configuring Multiple Contexts

#### Step 3

To allocate a physical interface:

```bash
allocate-interface physical_interface [mapped_name] [visible | invisible]
```

To allocate one or more subinterfaces:

```bash
allocate-interface physical_interface.subinterface[-physical_interface.subinterface] [mapped_name],[-mapped_name]] [visible | invisible]
```

**Example:**

```bash
hostname(config-ctx)# allocate-interface gigabitethernet0/1.100 int1
gigabitethernet0/1.200 int2
gigabitethernet0/2.300-gigabitethernet0/2.305 int3-int8
```

### Command Purpose

Specifies the interfaces you can use in the context. Do not include a space between the interface type and the port number.

Enter these commands multiple times to specify different ranges.

If you remove an allocation with the `no` form of this command, then any context commands that include this interface are removed from the running configuration.

You can assign the same interfaces to multiple contexts in routed mode, if desired.

The `mapped_name` is an alphanumeric alias for the interface that can be used within the context instead of the interface ID. If you do not specify a mapped name, the interface ID is used within the context. For security purposes, you might not want the context administrator to know which interfaces are being used by the context. A mapped name must start with a letter, end with a letter or digit, and have as interior characters only letters, digits, or an underscore. For example, you can use the following names:

```
int0, inta, int_0
```

If you specify a range of subinterfaces, you can specify a matching range of mapped names. Follow these guidelines for ranges:

- The mapped name must consist of an alphabetic portion followed by a numeric portion. The alphabetic portion of the mapped name must match for both ends of the range. For example, enter the following range:

  ```bash
  int0-int10
  ```

  If you enter `gig0/1.1-gig0/1.5 happy1-sad5`, for example, the command fails.

- The numeric portion of the mapped name must include the same quantity of numbers as the subinterface range. For example, both ranges include 100 interfaces:

  ```bash
  gigabitethernet0/0.100-gigabitethernet0/0.199 int1-int100
  ```

  If you enter `gig0/0.100-gig0/0.199 int1-int15`, for example, the command fails.

Specify `visible` to see the real interface ID in the `show interface` command if you set a mapped name. The default `invisible` keyword shows only the mapped name.
Chapter 6 Configuring Multiple Context Mode

Configuring Multiple Contexts

Examples

The following example sets the admin context to be “administrator,” creates a context called “administrator” on the internal flash memory, and then adds two contexts from an FTP server:

```bash
hostname(config)# admin-context administrator
hostname(config)# context administrator
hostname(config-ctx)# allocate-interface gigabitethernet0/0.1
hostname(config-ctx)# allocate-interface gigabitethernet0/1.1
hostname(config-ctx)# config-url flash:/admin.cfg
```
hostname(config-ctx)# context test
hostname(config-ctx)# allocate-interface gigabitethernet0/0.100 int1
hostname(config-ctx)# allocate-interface gigabitethernet0/0.102 int2
hostname(config-ctx)# allocate-interface gigabitethernet0/0.110-gigabitethernet0/0.115 int3-int8
hostname(config-ctx)# config-url ftp://user1:passw0rd@10.1.1.1/configlets/test.cfg
hostname(config-ctx)# member gold

hostname(config-ctx)# context sample
hostname(config-ctx)# allocate-interface gigabitethernet0/1.200 int1
hostname(config-ctx)# allocate-interface gigabitethernet0/1.212 int2
hostname(config-ctx)# allocate-interface gigabitethernet0/1.230-gigabitethernet0/1.235 int3-int8
hostname(config-ctx)# config-url ftp://user1:passw0rd@10.1.1.1/configlets/sample.cfg
hostname(config-ctx)# member silver

Automatically Assigning MAC Addresses to Context Interfaces

This section describes how to configure auto-generation of MAC addresses.

The MAC address is used to classify packets within a context. See the “Information About MAC Addresses” section on page 6-11 for more information, especially if you are upgrading from an earlier ASA version. See also the “Viewing Assigned MAC Addresses” section on page 6-33.

Guidelines

- When you configure a nameif command for the interface in a context, the new MAC address is generated immediately. If you enable this feature after you configure context interfaces, then MAC addresses are generated for all interfaces immediately after you enable it. If you disable this feature, the MAC address for each interface reverts to the default MAC address. For example, subinterfaces of GigabitEthernet 0/1 revert to using the MAC address of GigabitEthernet 0/1.

- In the rare circumstance that the generated MAC address conflicts with another private MAC address in your network, you can manually set the MAC address for the interface within the context. See the “Configuring the MAC Address and MTU” section on page 7-6 to manually set the MAC address.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>mac-address auto [prefix prefix]</td>
<td>Automatically assign private MAC addresses to each context interface. The prefix is a decimal value between 0 and 65535. This prefix is converted to a 4-digit hexadecimal number, and used as part of the MAC address. The prefix ensures that each ASASM uses unique MAC addresses, so you can have multiple ASASMs on a network segment, for example. See the “MAC Address Format” section for more information about how the prefix is used.</td>
</tr>
</tbody>
</table>

Example:
hostname(config)# mac-address auto prefix 19
Changing Between Contexts and the System Execution Space

If you log in to the system execution space (or the admin context using Telnet or SSH), you can change between contexts and perform configuration and monitoring tasks within each context. The running configuration that you edit in a configuration mode, or that is used in the copy or write commands, depends on your location. When you are in the system execution space, the running configuration consists only of the system configuration; when you are in a context, the running configuration consists only of that context. For example, you cannot view all running configurations (system plus all contexts) by entering the show running-config command. Only the current configuration displays.

To change between the system execution space and a context, or between contexts, see the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>changeto context name</code></td>
<td>Changes to a context. The prompt changes to the following:</td>
</tr>
<tr>
<td></td>
<td>hostname/name#</td>
</tr>
<tr>
<td><code>changeto system</code></td>
<td>Changes to the system execution space. The prompt changes to the</td>
</tr>
<tr>
<td></td>
<td>following:</td>
</tr>
<tr>
<td></td>
<td>hostname#</td>
</tr>
</tbody>
</table>

Managing Security Contexts

This section describes how to manage security contexts and includes the following topics:

- Removing a Security Context, page 6-22
- Changing the Admin Context, page 6-23
- Changing the Security Context URL, page 6-23
- Reloading a Security Context, page 6-24

Removing a Security Context

You can only remove a context by editing the system configuration. You cannot remove the current admin context, unless you remove all contexts using the clear context command.

Note

If you use failover, there is a delay between when you remove the context on the active unit and when the context is removed on the standby unit. You might see an error message indicating that the number of interfaces on the active and standby units are not consistent; this error is temporary and can be ignored.

Prerequisites

Perform this procedure in the system execution space.
Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>no context name</code></td>
<td>Removes a single context. All context commands are also removed.</td>
</tr>
<tr>
<td><code>clear context</code></td>
<td>Removes all contexts (including the admin context).</td>
</tr>
</tbody>
</table>

### Changing the Admin Context

The system configuration does not include any network interfaces or network settings for itself; rather, when the system needs to access network resources (such as downloading the contexts from the server), it uses one of the contexts that is designated as the admin context.

The admin context is just like any other context, except that when a user logs in to the admin context, then that user has system administrator rights and can access the system and all other contexts. The admin context is not restricted in any way, and can be used as a regular context. However, because logging into the admin context grants you administrator privileges over all contexts, you might need to restrict access to the admin context to appropriate users.

### Guidelines

You can set any context to be the admin context, as long as the configuration file is stored in the internal flash memory.

### Prerequisites

Perform this procedure in the system execution space.

### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>admin-context</code></td>
<td>Sets the admin context. Any remote management sessions, such as Telnet,</td>
</tr>
<tr>
<td><code>context_name</code></td>
<td>SSH, or HTTPS, that are connected to the admin context are terminated.</td>
</tr>
<tr>
<td></td>
<td>You must reconnect to the new admin context.</td>
</tr>
</tbody>
</table>

**Example:**
```
hostname(config)# admin-context administrator
```

**Note**
A few system commands, including `ntp server`, identify an interface name that belongs to the admin context. If you change the admin context, and that interface name does not exist in the new admin context, be sure to update any system commands that refer to the interface.

### Changing the Security Context URL

This section describes how to change the context URL.
Guidelines

- You cannot change the security context URL without reloading the configuration from the new URL. The ASASM merges the new configuration with the current running configuration.
- Reentering the same URL also merges the saved configuration with the running configuration.

A merge adds any new commands from the new configuration to the running configuration.
- If the configurations are the same, no changes occur.
- If commands conflict or if commands affect the running of the context, then the effect of the merge depends on the command. You might get errors, or you might have unexpected results. If the running configuration is blank (for example, if the server was unavailable and the configuration was never downloaded), then the new configuration is used.

If you do not want to merge the configurations, you can clear the running configuration, which disrupts any communications through the context, and then reload the configuration from the new URL.

Prerequisites

Perform this procedure in the system execution space.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td>(Optional, if you do not want to perform a</td>
<td>Changes to the context and clears its configuration. If you want to</td>
</tr>
<tr>
<td>merge)</td>
<td>perform a merge, skip to Step 2.</td>
</tr>
<tr>
<td><code>changeto context name</code></td>
<td></td>
</tr>
<tr>
<td><code>clear configure all</code></td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>hostname(config)# changeto context ctx1</code></td>
<td></td>
</tr>
<tr>
<td><code>hostname/ctx1(config)# clear configure all</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Changes to the system execution space.</td>
</tr>
<tr>
<td><code>changeto system</code></td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>hostname/ctx1(config)# changeto system</code></td>
<td></td>
</tr>
<tr>
<td><code>hostname(config)#</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>Enters the context configuration mode for the context you want to</td>
</tr>
<tr>
<td><code>context name</code></td>
<td>change.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>hostname(config)# context ctx1</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>Enters the new URL. The system immediately loads the context so that it is running.</td>
</tr>
<tr>
<td><code>config-url new_url</code></td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>hostname(config)# config-url ftp://user1:passw0rd@10.1.1.1/configlets/ctx1.cfg</code></td>
<td></td>
</tr>
</tbody>
</table>
Reloading a Security Context

You can reload the context in two ways:

- Clear the running configuration and then import the startup configuration.
  This action clears most attributes associated with the context, such as connections and NAT tables.
- Remove the context from the system configuration.
  This action clears additional attributes, such as memory allocation, which might be useful for troubleshooting. However, to add the context back to the system requires you to respecify the URL and interfaces.

This section includes the following topics:

- Reloading by Clearing the Configuration, page 6-25
- Reloading by Removing and Re-adding the Context, page 6-25

Reloading by Clearing the Configuration

To reload the context by clearing the context configuration, and reloading the configuration from the URL, perform the following steps.

**Detailed Steps**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>changeto context name</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname(config)# changeto context ctx1 hostname/ctx1(config)#</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>clear configure all</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname/ctx1(config)# clear configure all</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>copy startup-config running-config</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname/ctx1(config)# copy startup-config running-config</td>
</tr>
</tbody>
</table>

Reloading by Removing and Re-adding the Context

To reload the context by removing the context and then re-adding it, perform the steps in the following sections:

1. “Removing a Security Context” section on page 6-22
2. “Configuring a Security Context” section on page 6-17
Monitoring Security Contexts

This section describes how to view and monitor context information and includes the following topics:

- Viewing Context Information, page 6-26
- Viewing Resource Allocation, page 6-27
- Viewing Resource Usage, page 6-30
- Monitoring SYN Attacks in Contexts, page 6-31
- Viewing Assigned MAC Addresses, page 6-33

Viewing Context Information

From the system execution space, you can view a list of contexts including the name, allocated interfaces, and configuration file URL.

From the system execution space, view all contexts by entering the following command:

```
hostname# show context
```

The following is sample output from the `show context` command. The following sample output shows three contexts:

```
hostname# show context

  Context Name  Interfaces          URL
  *admin       GigabitEthernet0/1.100 disk0:/admin.cfg
                GigabitEthernet0/1.101
  contexta     GigabitEthernet0/1.200 disk0:/contexta.cfg
                GigabitEthernet0/1.201
  contextb     GigabitEthernet0/1.300 disk0:/contextb.cfg
                GigabitEthernet0/1.301

Total active Security Contexts: 3
```

Table 6-2 shows each field description.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Context Name</td>
<td>Lists all context names. The context name with the asterisk (*) is the admin context.</td>
</tr>
<tr>
<td>Interfaces</td>
<td>The interfaces assigned to the context.</td>
</tr>
<tr>
<td>URL</td>
<td>The URL from which the ASASM loads the context configuration.</td>
</tr>
</tbody>
</table>
The following is sample output from the `show context detail` command:

```
hostname# show context detail

Context "admin", has been created, but initial ACL rules not complete
  Config URL: disk0:/admin.cfg
  Real Interfaces: Management0/0
  Mapped Interfaces: Management0/0
  Flags: 0x00000013, ID: 1

Context "ctx", has been created, but initial ACL rules not complete
  Config URL: ctx.cfg
  Real Interfaces: GigabitEthernet0/0.10, GigabitEthernet0/1.20,
  GigabitEthernet0/2.30
  Mapped Interfaces: int1, int2, int3
  Flags: 0x00000011, ID: 2

Context "system", is a system resource
  Config URL: startup-config
  Real Interfaces: Control0/0, GigabitEthernet0/0,
  GigabitEthernet0/0.10, GigabitEthernet0/1, GigabitEthernet0/1.10,
  GigabitEthernet0/1.20, GigabitEthernet0/2, GigabitEthernet0/2.30,
  GigabitEthernet0/3, Management0/0, Management0/0.1
  Flags: 0x00000019, ID: 257

Context "null", is a system resource
  Config URL: ... null ...
  Real Interfaces: 
  Mapped Interfaces: 
  Flags: 0x00000009, ID: 258
```

See the command reference for more information about the `detail` output.

The following is sample output from the `show context count` command:

```
hostname# show context count
Total active contexts: 2
```

### Viewing Resource Allocation

From the system execution space, you can view the allocation for each resource across all classes and class members.

To view the resource allocation, enter the following command:

```
show resource allocation [detail]
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show resource allocation [detail]</code></td>
<td>Shows the resource allocation. This command shows the resource allocation, but does not show the actual resources being used. See the “Viewing Resource Usage” section on page 6-30 for more information about actual resource usage. The <code>detail</code> argument shows additional information. See the following sample outputs for more information.</td>
</tr>
</tbody>
</table>

The following sample output shows the total allocation of each resource as an absolute value and as a percentage of the available system resources:

```
hostname# show resource allocation
```
Monitoring Security Contexts

Table 6-3 shows each field description.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resource</td>
<td>The name of the resource that you can limit.</td>
</tr>
<tr>
<td>Total</td>
<td>The total amount of the resource that is allocated across all contexts.</td>
</tr>
<tr>
<td></td>
<td>The amount is an absolute number of concurrent instances or instances per second. If you specified a percentage in the class definition, the ASASM converts the percentage to an absolute number for this display.</td>
</tr>
<tr>
<td>% of Avail</td>
<td>The percentage of the total system resources that is allocated across all contexts, if the resource has a hard system limit. If a resource does not have a system limit, this column shows N/A.</td>
</tr>
</tbody>
</table>

The following is sample output from the `show resource allocation detail` command:

```
hostname# show resource allocation detail
Resource Origin:
  A  Value was derived from the resource 'all'
  C  Value set in the definition of this class
  D  Value set in default class

<table>
<thead>
<tr>
<th>Resource</th>
<th>Class</th>
<th>Mbrs</th>
<th>Origin</th>
<th>Limit</th>
<th>Total</th>
<th>Total %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conns [rate]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>default</td>
<td>all</td>
<td>CA</td>
<td>unlimited</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>gold</td>
<td>1</td>
<td>C</td>
<td>34000</td>
<td>34000</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>silver</td>
<td>1</td>
<td>CA</td>
<td>17000</td>
<td>17000</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>bronze</td>
<td>0</td>
<td>CA</td>
<td>8500</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>All Contexts: 3</td>
<td></td>
<td></td>
<td>51000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Inspects [rate]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>default</td>
<td>all</td>
<td>CA</td>
<td>unlimited</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>gold</td>
<td>1</td>
<td>DA</td>
<td>unlimited</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>silver</td>
<td>1</td>
<td>CA</td>
<td>10000</td>
<td>10000</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>bronze</td>
<td>0</td>
<td>CA</td>
<td>5000</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>All Contexts: 3</td>
<td></td>
<td></td>
<td>10000</td>
<td></td>
<td>N/A</td>
</tr>
<tr>
<td>Syslogs [rate]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>default</td>
<td>all</td>
<td>CA</td>
<td>unlimited</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>gold</td>
<td>1</td>
<td>C</td>
<td>6000</td>
<td>6000</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>silver</td>
<td>1</td>
<td>CA</td>
<td>3000</td>
<td>3000</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>bronze</td>
<td>0</td>
<td>CA</td>
<td>1500</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>All Contexts: 3</td>
<td></td>
<td></td>
<td>9000</td>
<td></td>
<td>N/A</td>
</tr>
<tr>
<td>Conns</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>default</td>
<td>all</td>
<td>CA</td>
<td>unlimited</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>gold</td>
<td>1</td>
<td>C</td>
<td>200000</td>
<td>200000</td>
<td>20.00%</td>
</tr>
<tr>
<td></td>
<td>silver</td>
<td>1</td>
<td>CA</td>
<td>100000</td>
<td>100000</td>
<td>10.00%</td>
</tr>
<tr>
<td></td>
<td>bronze</td>
<td>0</td>
<td>CA</td>
<td>50000</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>All Contexts: 3</td>
<td></td>
<td></td>
<td>300000</td>
<td></td>
<td>30.00%</td>
</tr>
<tr>
<td>Hosts</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>default</td>
<td>all</td>
<td>CA</td>
<td>unlimited</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>gold</td>
<td>1</td>
<td>DA</td>
<td>unlimited</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```
### Table 6-4 shows each field description.

**Table 6-4 show resource allocation detail Fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resource</td>
<td>The name of the resource that you can limit.</td>
</tr>
</tbody>
</table>
| Class         | The name of each class, including the default class.  
                | The All contexts field shows the total values across all classes.          |
| Mmbrs         | The number of contexts assigned to each class.                              |
| Origin        | The origin of the resource limit, as follows:                              |
|               | • A—You set this limit with the all option, instead of as an individual resource. |
|               | • C—This limit is derived from the member class.                           |
|               | • D—This limit was not defined in the member class, but was derived from the default class. For a context assigned to the default class, the value will be “C” instead of “D.” |
|               | The ASASM can combine “A” with “C” or “D.”                                 |
| Limit         | The limit of the resource per context, as an absolute number. If you specified a percentage in the class definition, the ASASM converts the percentage to an absolute number for this display. |
| Total         | The total amount of the resource that is allocated across all contexts in the class. The amount is an absolute number of concurrent instances or instances per second. If the resource is unlimited, this display is blank. |
| % of Avail    | The percentage of the total system resources that is allocated across all contexts in the class. If the resource is unlimited, this display is blank. If the resource does not have a system limit, then this column shows N/A. |
Viewing Resource Usage

From the system execution space, you can view the resource usage for each context and display the system resource usage.

From the system execution space, view the resource usage for each context by entering the following command:

```
hostname# show resource usage context admin
```

The following is sample output from the `show resource usage context` command, which shows the resource usage for the admin context:

```
Resource     Current  Peak  Limit  Denied  Context
Telnet       1        1      5      0  admin
Conns        44       55    N/A    0  admin
Hosts        45       56    N/A    0  admin
```

Command Purpose

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show resource usage [context context_name [top n</td>
<td>all</td>
</tr>
<tr>
<td>current—Shows the active concurrent instances or the current rate of the resource. denied—Shows the number of instances that were denied because they exceeded the resource limit shown in the Limit column. peak—Shows the peak concurrent instances, or the peak rate of the resource since the statistics were last cleared, either using the clear resource usage command or because the device rebooted. all—(Default) Shows all statistics. The count_threshold sets the number above which resources are shown. The default is 1. If the usage of the resource is below the number you set, then the resource is not shown. If you specify all for the counter name, then the count_threshold applies to the current usage.</td>
<td></td>
</tr>
<tr>
<td>The following is sample output from the show resource usage context command, which shows the resource usage for the admin context:</td>
<td></td>
</tr>
</tbody>
</table>

Note To show all resources, set the count_threshold to 0.
The following is sample output from the `show resource usage summary` command, which shows the resource usage for all contexts and all resources. This sample shows the limits for 6 contexts.

```
hostname# show resource usage summary
```

<table>
<thead>
<tr>
<th>Resource</th>
<th>Current</th>
<th>Peak</th>
<th>Limit</th>
<th>Denied</th>
<th>Context</th>
</tr>
</thead>
<tbody>
<tr>
<td>Syslogs [rate]</td>
<td>1743</td>
<td>2132</td>
<td>N/A</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>Conns</td>
<td>584</td>
<td>763</td>
<td>280000(S)</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>Xlates</td>
<td>8526</td>
<td>8966</td>
<td>N/A</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>Hosts</td>
<td>254</td>
<td>254</td>
<td>N/A</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>Conns [rate]</td>
<td>270</td>
<td>535</td>
<td>N/A</td>
<td>1704</td>
<td>Summary</td>
</tr>
<tr>
<td>Inspects [rate]</td>
<td>270</td>
<td>535</td>
<td>N/A</td>
<td>0</td>
<td>Summary</td>
</tr>
</tbody>
</table>

S = System: Combined context limits exceed the system limit; the system limit is shown.

The following is sample output from the `show resource usage summary` command, which shows the limits for 25 contexts. Because the context limit for Telnet and SSH connections is 5 per context, then the combined limit is 125. The system limit is only 100, so the system limit is shown.

```
hostname# show resource usage summary
```

<table>
<thead>
<tr>
<th>Resource</th>
<th>Current</th>
<th>Peak</th>
<th>Limit</th>
<th>Denied</th>
<th>Context</th>
</tr>
</thead>
<tbody>
<tr>
<td>Telnet</td>
<td>1</td>
<td>1</td>
<td>100(S)</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>SSH</td>
<td>2</td>
<td>2</td>
<td>100(S)</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>Conns</td>
<td>56</td>
<td>90</td>
<td>N/A</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>Hosts</td>
<td>89</td>
<td>102</td>
<td>N/A</td>
<td>0</td>
<td>Summary</td>
</tr>
</tbody>
</table>

S = System: Combined context limits exceed the system limit; the system limit is shown.

The following is sample output from the `show resource usage system` command, which shows the resource usage for all contexts, but it shows the system limit instead of the combined context limits. The `counter all 0` option is used to show resources that are not currently in use. The Denied statistics indicate how many times the resource was denied due to the system limit, if available.

```
hostname# show resource usage system counter all 0
```

<table>
<thead>
<tr>
<th>Resource</th>
<th>Current</th>
<th>Peak</th>
<th>Limit</th>
<th>Denied</th>
<th>Context</th>
</tr>
</thead>
<tbody>
<tr>
<td>Telnet</td>
<td>0</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>System</td>
</tr>
<tr>
<td>SSH</td>
<td>0</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>System</td>
</tr>
<tr>
<td>ASDM</td>
<td>0</td>
<td>0</td>
<td>32</td>
<td>0</td>
<td>System</td>
</tr>
<tr>
<td>Syslogs [rate]</td>
<td>1</td>
<td>18</td>
<td>N/A</td>
<td>0</td>
<td>System</td>
</tr>
<tr>
<td>Conns</td>
<td>0</td>
<td>1</td>
<td>280000</td>
<td>0</td>
<td>System</td>
</tr>
<tr>
<td>Xlates</td>
<td>0</td>
<td>0</td>
<td>N/A</td>
<td>0</td>
<td>System</td>
</tr>
<tr>
<td>Hosts</td>
<td>0</td>
<td>2</td>
<td>N/A</td>
<td>0</td>
<td>System</td>
</tr>
<tr>
<td>Conns [rate]</td>
<td>1</td>
<td>1</td>
<td>N/A</td>
<td>0</td>
<td>System</td>
</tr>
<tr>
<td>Inspects [rate]</td>
<td>0</td>
<td>0</td>
<td>N/A</td>
<td>0</td>
<td>System</td>
</tr>
</tbody>
</table>

**Monitoring SYN Attacks in Contexts**

The ASASM prevents SYN attacks using TCP Intercept. TCP Intercept uses the SYN cookies algorithm to prevent TCP SYN-flooding attacks. A SYN-flooding attack consists of a series of SYN packets usually originating from spoofed IP addresses. The constant flood of SYN packets keeps the server SYN queue full, which prevents it from servicing connection requests. When the embryonic connection threshold of a connection is crossed, the ASASM acts as a proxy for the server and generates a SYN-ACK response to the client SYN request. When the ASASM receives an ACK back from the client, it can then authenticate the client and allow the connection to the server.

Monitor SYN attacks using the following commands:
The following is sample output from the `show perfmon` command that shows the rate of TCP intercepts for a context called admin.

```
hostname/admin# show perfmon
Context: admin
PERFMON STATS: Current Average
Xlates 0/s 0/s
Connections 0/s 0/s
TCP Conn 0/s 0/s
UDP Conn 0/s 0/s
URL Access 0/s 0/s
URL Server Req 0/s 0/s
WebSns Req 0/s 0/s
TCP Fixup 0/s 0/s
HTTP Fixup 0/s 0/s
FTP Fixup 0/s 0/s
AAA Authen 0/s 0/s
AAA Author 0/s 0/s
AAA Account 0/s 0/s
TCP Intercept 322779/s 322779/s
```

The following is sample output from the `show resource usage detail` command that shows the amount of resources being used by TCP Intercept for individual contexts. (Sample text in italics shows the TCP intercept information.)

```
hostname(config)# show resource usage detail
Resource Current Peak Limit Denied Context
memory 843732 847288 unlimited 0 admin
chunk:channels 14 15 unlimited 0 admin
chunk:fixup 15 15 unlimited 0 admin
chunk:hole 1 1 unlimited 0 admin
chunk:ip-users 10 10 unlimited 0 admin
chunk:list-elem 21 21 unlimited 0 admin
chunk:list-hdr 3 4 unlimited 0 admin
chunk:route 2 2 unlimited 0 admin
chunk:static 1 1 unlimited 0 admin
tcp-intercepts 328787 803610 unlimited 0 admin
np-statics 3 3 unlimited 0 admin
statics 1 1 unlimited 0 admin
ace-rules 1 1 unlimited 0 admin
console-access-rul 2 2 unlimited 0 admin
fixup-rules 14 15 unlimited 0 admin
memory 959872 960000 unlimited 0 c1
chunk:channels 15 16 unlimited 0 c1
chunk:dbtrace 1 1 unlimited 0 c1
chunk:fixup 15 15 unlimited 0 c1
chunk:global 1 1 unlimited 0 c1
chunk:hole 2 2 unlimited 0 c1
chunk:ip-users 10 10 unlimited 0 c1
chunk:udp-ctrl-blk 1 1 unlimited 0 c1
chunk:list-elem 24 24 unlimited 0 c1
chunk:list-hdr 5 6 unlimited 0 c1
```
The following sample output shows the resources being used by TCP intercept for the entire system. (Sample text in italics shows the TCP intercept information.)

```
hostname(config)# show resource usage summary detail
<table>
<thead>
<tr>
<th>Resource</th>
<th>Current</th>
<th>Peak</th>
<th>Limit</th>
<th>Denied</th>
<th>Context</th>
</tr>
</thead>
<tbody>
<tr>
<td>memory</td>
<td>238421312</td>
<td>238434336</td>
<td>unlimited</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>chunk:channels</td>
<td>46</td>
<td>48</td>
<td>unlimited</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>chunk:dbgtrace</td>
<td>4</td>
<td>4</td>
<td>unlimited</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>chunk:fixup</td>
<td>45</td>
<td>45</td>
<td>unlimited</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>chunk:global</td>
<td>1</td>
<td>1</td>
<td>unlimited</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>chunk:hole</td>
<td>3</td>
<td>3</td>
<td>unlimited</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>chunk:ip-users</td>
<td>24</td>
<td>24</td>
<td>unlimited</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>chunk:udp-ctrl-blk</td>
<td>1</td>
<td>1</td>
<td>unlimited</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>chunk:list-elem</td>
<td>1059</td>
<td>1059</td>
<td>unlimited</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>chunk:list-hdr</td>
<td>10</td>
<td>11</td>
<td>unlimited</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>chunk:nat</td>
<td>1</td>
<td>1</td>
<td>unlimited</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>chunk:route</td>
<td>5</td>
<td>5</td>
<td>unlimited</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>chunk:static</td>
<td>2</td>
<td>2</td>
<td>unlimited</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>block:16384</td>
<td>510</td>
<td>885</td>
<td>unlimited</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>block:2048</td>
<td>32</td>
<td>35</td>
<td>unlimited</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>tcp-intercept-rate</td>
<td>341306</td>
<td>811579</td>
<td>unlimited</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>globals</td>
<td>1</td>
<td>1</td>
<td>unlimited</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>np-statics</td>
<td>6</td>
<td>6</td>
<td>unlimited</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>statics</td>
<td>2</td>
<td>2</td>
<td>N/A</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>nats</td>
<td>1</td>
<td>1</td>
<td>N/A</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>ace-rules</td>
<td>3</td>
<td>3</td>
<td>N/A</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>console-access-rul</td>
<td>4</td>
<td>4</td>
<td>N/A</td>
<td>0</td>
<td>Summary</td>
</tr>
<tr>
<td>fixup-rules</td>
<td>43</td>
<td>44</td>
<td>N/A</td>
<td>0</td>
<td>Summary</td>
</tr>
</tbody>
</table>
```

**Viewing Assigned MAC Addresses**

You can view auto-generated MAC addresses within the system configuration or within the context. This section includes the following topics:

- Viewing MAC Addresses in the System Configuration, page 6-34
- Viewing MAC Addresses Within a Context, page 6-35
Viewing MAC Addresses in the System Configuration

This section describes how to view MAC addresses in the system configuration.

Guidelines

If you manually assign a MAC address to an interface, but also have auto-generation enabled, the auto-generated address continues to show in the configuration even though the manual MAC address is the one that is in use. If you later remove the manual MAC address, the auto-generated one shown will be used.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show running-config all context [name]</td>
<td>Shows the assigned MAC addresses from the system execution space. The all option is required to view the assigned MAC addresses. Although this command is user-configurable in global configuration mode only, the <strong>mac-address auto</strong> command appears as a read-only entry in the configuration for each context along with the assigned MAC address. Only allocated interfaces that are configured with a <strong>nameif</strong> command within the context have a MAC address assigned.</td>
</tr>
</tbody>
</table>

Examples

The following output from the **show running-config all context admin** command shows the primary and standby MAC address assigned to the Management0/0 interface:

```
hostname# show running-config all context admin
context admin
    allocate-interface Management0/0
    mac-address auto Management0/0 a24d.0000.1440 a24d.0000.1441
    config-url disk0:/admin.cfg
```

The following output from the **show running-config all context** command shows all the MAC addresses (primary and standby) for all context interfaces. Note that because the GigabitEthernet0/0 and GigabitEthernet0/1 main interfaces are not configured with a **nameif** command inside the contexts, no MAC addresses have been generated for them.

```
hostname# show running-config all context

admin-context admin
context admin
    allocate-interface Management0/0
    mac-address auto Management0/0 a2d2.0400.125a a2d2.0400.125b
    config-url disk0:/admin.cfg

context CTX1
    allocate-interface GigabitEthernet0/0
    allocate-interface GigabitEthernet0/0.1-GigabitEthernet0/0.5
    mac-address auto GigabitEthernet0/0.1 a2d2.0400.11bc a2d2.0400.11bd
    mac-address auto GigabitEthernet0/0.2 a2d2.0400.11c0 a2d2.0400.11c1
    mac-address auto GigabitEthernet0/0.3 a2d2.0400.11c4 a2d2.0400.11c5
    mac-address auto GigabitEthernet0/0.4 a2d2.0400.11c8 a2d2.0400.11c9
```
Viewing MAC Addresses Within a Context

This section describes how to view MAC addresses within a context.

**Detailed Steps**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`show interface</td>
<td>include (Interface)</td>
</tr>
</tbody>
</table>

**Examples**

For example:

```
hostname/context# show interface | include (Interface)|(MAC)
```

Interface GigabitEthernet1/1.1 "g1/1.1", is down, line protocol is down
MAC address a201.0101.0600, MTU 1500
Interface GigabitEthernet1/1.2 "g1/1.2", is down, line protocol is down
MAC address a201.0102.0600, MTU 1500
Interface GigabitEthernet1/1.3 "g1/1.3", is down, line protocol is down
MAC address a201.0103.0600, MTU 1500
...

**Note**

The `show interface` command shows the MAC address in use; if you manually assign a MAC address and also have auto-generation enabled, then you can only view the unused auto-generated address from within the system configuration.
Configuration Examples for Multiple Context Mode

The following example:

- Automatically sets the MAC addresses in contexts.
- Sets the default class limit for conns to 10 percent instead of unlimited.
- Creates a gold resource class.
- Sets the admin context to be “administrator.”
- Creates a context called “administrator” on the internal flash memory to be part of the default resource class.
- Adds two contexts from an FTP server as part of the gold resource class.

```
hostname(config)# mac-address auto prefix 19
hostname(config)# class default
hostname(config-class)# limit-resource conns 10%

hostname(config)# class gold
hostname(config-class)# limit-resource mac-addresses 10000
hostname(config-class)# limit-resource conns 15%
hostname(config-class)# limit-resource rate conns 1000
hostname(config-class)# limit-resource rate inspects 500
hostname(config-class)# limit-resource hosts 9000
hostname(config-class)# limit-resource asdm 5
hostname(config-class)# limit-resource ssh 5
hostname(config-class)# limit-resource rate syslogs 5000
hostname(config-class)# limit-resource telnet 5
hostname(config-class)# limit-resource xlates 36000

hostname(config)# admin-context administrator
hostname(config)# context administrator
hostname(config-ctx)# allocate-interface gigabitethernet0/0.1
hostname(config-ctx)# allocate-interface gigabitethernet0/0.110-gigabitethernet0/0.115 int3-int8
hostname(config-ctx)# config-url flash:/admin.cfg

hostname(config-ctx)# context test
hostname(config-ctx)# allocate-interface gigabitethernet0/0.100 int1
hostname(config-ctx)# allocate-interface gigabitethernet0/0.102 int2
hostname(config-ctx)# allocate-interface gigabitethernet0/0.110-gigabitethernet0/0.115 int3-int8
hostname(config-ctx)# config-url ftp://user1:passw0rd@10.1.1.1/configlets/test.cfg
hostname(config-ctx)# member gold

hostname(config-ctx)# context sample
hostname(config-ctx)# allocate-interface gigabitethernet0/1.200 int1
hostname(config-ctx)# allocate-interface gigabitethernet0/1.212 int2
hostname(config-ctx)# allocate-interface gigabitethernet0/1.230-gigabitethernet0/1.235 int3-int8
hostname(config-ctx)# config-url ftp://user1:passw0rd@10.1.1.1/configlets/sample.cfg
hostname(config-ctx)# member gold
```
Feature History for Multiple Context Mode

Table 6-5 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multiple security contexts</td>
<td>7.0(1)</td>
<td>Multiple context mode was introduced.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>We introduced the following commands: <code>context</code>, <code>mode</code>, and <code>class</code>.</td>
</tr>
<tr>
<td>Automatic MAC address assignment</td>
<td>7.2(1)</td>
<td>Automatic assignment of MAC address to context interfaces was introduced.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>We introduced the following command: <code>mac-address auto</code>.</td>
</tr>
<tr>
<td>Resource management</td>
<td>7.2(1)</td>
<td>Resource management was introduced.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>We introduced the following commands: <code>class</code>, <code>limit-resource</code>, and <code>member</code>.</td>
</tr>
<tr>
<td>Virtual sensors for IPS</td>
<td>8.0(2)</td>
<td>The AIP SSM running IPS software Version 6.0 and above can run multiple virtual sensors, which means you can configure multiple security policies</td>
</tr>
<tr>
<td></td>
<td></td>
<td>on the AIP SSM. You can assign each context or single mode ASASM to one or more virtual sensors, or you can assign multiple security contexts to</td>
</tr>
<tr>
<td></td>
<td></td>
<td>the same virtual sensor.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>We introduced the following command: <code>allocate-ips</code>.</td>
</tr>
<tr>
<td>Automatic MAC address assignment enhancements</td>
<td>8.0(5)/8.2(2)</td>
<td>The MAC address format was changed to use a prefix, to use a fixed starting value (A2), and to use a different scheme for the primary and secondary</td>
</tr>
<tr>
<td></td>
<td></td>
<td>unit MAC addresses in a failover pair. The MAC addresses are also now persistent across reloads. The command parser now checks if auto-generation</td>
</tr>
<tr>
<td></td>
<td></td>
<td>is enabled; if you want to also manually assign a MAC address, you cannot start the manual MAC address with A2.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>We modified the following command: <code>mac-address auto prefix</code>.</td>
</tr>
<tr>
<td>Maximum contexts increased for the ASA 5550</td>
<td>8.4(1)</td>
<td>The maximum security contexts for the ASA 5550 was increased from 50 to 100. The maximum for the ASA 5580 was increased from 50 to 250.</td>
</tr>
</tbody>
</table>
### Feature History for Multiple Context Mode

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Automatic MAC address assignment enabled by default</td>
<td>8.5(1)</td>
<td>Automatic MAC address assignment is now enabled by default. We modified the following command: <code>mac-address auto</code>.</td>
</tr>
<tr>
<td>Automatic generation of a MAC address prefix for the <code>mac-address auto</code> command</td>
<td>8.6(1)</td>
<td>In multiple context mode, the ASASM now converts the automatic MAC address generation configuration to use a default prefix. The ASASM auto-generates the prefix based on the last two bytes of the interface MAC address. This conversion happens automatically when you reload, or if you reenable MAC address generation. The prefix method of generation provides many benefits, including a better guarantee of unique MAC addresses on a segment. You can view the auto-generated prefix by entering the <code>show running-config mac-address</code> command. If you want to change the prefix, you can reconfigure the feature with a custom prefix. The legacy method of MAC address generation is no longer available. <strong>Note</strong> To maintain hitless upgrade for failover pairs, the ASASM does not convert the MAC address method in an existing configuration upon a reload if failover is enabled. However, we strongly recommend that you manually change to the prefix method of generation when using failover. After upgrading, to use the prefix method of MAC address generation, reenable MAC address generation to use the default prefix. We modified the following command: <code>mac-address auto</code>.</td>
</tr>
</tbody>
</table>
PART 3

Configuring Interfaces
CHAPTER 7

Configuring Interfaces (Routed Mode)

This chapter includes tasks to complete the interface configuration for all models in routed firewall mode. This chapter includes the following sections:

- Information About Completing Interface Configuration in Routed Mode, page 7-1
- Licensing Requirements for Completing Interface Configuration in Routed Mode, page 7-2
- Guidelines and Limitations, page 7-3
- Default Settings, page 7-3
- Completing Interface Configuration in Routed Mode, page 7-4
- Turning Off and Turning On Interfaces, page 7-14
- Monitoring Interfaces, page 7-14
- Feature History for Interfaces in Routed Mode, page 7-15
- Feature History for Interfaces in Routed Mode, page 7-15

Note
For multiple context mode, complete the tasks in this section in the context execution space. Enter the change to context name command to change to the context you want to configure.

Information About Completing Interface Configuration in Routed Mode

This section includes the following topics:

- Security Levels, page 7-1
- Dual IP Stack (IPv4 and IPv6), page 7-2

Security Levels

Each interface must have a security level from 0 (lowest) to 100 (highest). For example, you should assign your most secure network, such as the inside host network, to level 100. While the outside network connected to the Internet can be level 0. Other networks, such as DMZs can be in between. You can assign interfaces to the same security level. See the “Allowing Same Security Level Communication” section on page 7-12 for more information.
The level controls the following behavior:

- Network access—By default, there is an implicit permit from a higher security interface to a lower security interface (outbound). Hosts on the higher security interface can access any host on a lower security interface. You can limit access by applying an access list to the interface.

  If you enable communication for same security interfaces (see the “Allowing Same Security Level Communication” section on page 7-12), there is an implicit permit for interfaces to access other interfaces on the same security level or lower.

- Inspection engines—Some application inspection engines are dependent on the security level. For same security interfaces, inspection engines apply to traffic in either direction.
  - NetBIOS inspection engine—Applied only for outbound connections.
  - SQL*Net inspection engine—If a control connection for the SQL*Net (formerly OraServ) port exists between a pair of hosts, then only an inbound data connection is permitted through the ASASM.

- Filtering—HTTP(S) and FTP filtering applies only for outbound connections (from a higher level to a lower level).

  If you enable communication for same security interfaces, you can filter traffic in either direction.

- established command—This command allows return connections from a lower security host to a higher security host if there is already an established connection from the higher level host to the lower level host.

  If you enable communication for same security interfaces, you can configure established commands for both directions.

### Dual IP Stack (IPv4 and IPv6)

The ASASM supports the configuration of both IPv6 and IPv4 on an interface. You do not need to enter any special commands to do so; simply enter the IPv4 configuration commands and IPv6 configuration commands as you normally would. Make sure you configure a default route for both IPv4 and IPv6.

### Licensing Requirements for Completing Interface Configuration in Routed Mode

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASASM</td>
<td>VLANs:</td>
</tr>
<tr>
<td></td>
<td>Base License: 1000</td>
</tr>
</tbody>
</table>
Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

Context Mode Guidelines

- For the ASASM in multiple context mode, configure switch ports and VLANs on the switch, and then assign VLANs to the ASASM according to Chapter 2, “Configuring the Switch for Use with the ASA Services Module.”
- In multiple context mode, you can only configure context interfaces that you already assigned to the context in the system configuration according to the “Configuring Multiple Contexts” section on page 6-14.
- PPPoE is not supported in multiple context mode.

Firewall Mode Guidelines

Supported in routed firewall mode. For transparent mode, see Chapter 8, “Configuring Interfaces (Transparent Mode).”

Failover Guidelines

Do not finish configuring failover interfaces with the procedures in this chapter. See the “Configuring Active/Standby Failover” section on page 50-7 or the “Configuring Active/Active Failover” section on page 51-8 to configure the failover and state links. In multiple context mode, failover interfaces are configured in the system configuration.

IPv6 Guidelines

Supports IPv6.

VLAN ID Guidelines for the ASASM

You can add any VLAN ID to the configuration, but only VLANs that are assigned to the ASASM by the switch can pass traffic. To view all VLANs assigned to the ASASM, use the `show vlan` command.

If you add an interface for a VLAN that is not yet assigned to the ASASM by the switch, the interface will be in the down state. When you assign the VLAN to the ASASM, the interface changes to an up state. See the `show interface` command for more information about interface states.

Default Settings

This section lists default settings for interfaces if you do not have a factory default configuration. For information about the factory default configurations, see the “Working with the Configuration” section on page 3-11.

Default Security Level

The default security level is 0. If you name an interface “inside” and you do not set the security level explicitly, then the ASASM sets the security level to 100.

**Note**

If you change the security level of an interface, and you do not want to wait for existing connections to time out before the new security information is used, you can clear the connections using the `clear local-host` command.
Default State of Interfaces for the ASASM

- In single mode or in the system execution space, VLAN interfaces are enabled by default.
- In multiple context mode, all allocated interfaces are enabled by default, no matter what the state of the interface is in the system execution space. However, for traffic to pass through the interface, the interface also has to be enabled in the system execution space. If you shut down an interface in the system execution space, then that interface is down in all contexts that share it.

Jumbo Frame Support

By default, the ASASM supports jumbo frames. Just configure the MTU for the desired packet size according to the “Configuring the MAC Address and MTU” section on page 7-6.

Completing Interface Configuration in Routed Mode

This section includes the following topics:
- Task Flow for Completing Interface Configuration, page 7-4
- Configuring General Interface Parameters, page 7-4
- Configuring the MAC Address and MTU, page 7-6
- Configuring IPv6 Addressing, page 7-8
- Allowing Same Security Level Communication, page 7-12

Task Flow for Completing Interface Configuration

Step 1
Set up your interfaces depending on your model:
- ASASM—Chapter 2, “Configuring the Switch for Use with the ASA Services Module.”

Step 2
(Multiple context mode) Allocate interfaces to the context according to the “Configuring Multiple Contexts” section on page 6-14.

Step 3
(Multiple context mode) Enter the `changeto context name` command to change to the context you want to configure. Configure general interface parameters, including the interface name, security level, and IPv4 address. See the “Configuring General Interface Parameters” section on page 7-4.

Step 4
(Optional) Configure the MAC address and the MTU. See the “Configuring the MAC Address and MTU” section on page 7-6.

Step 5
(Optional) Configure IPv6 addressing. See the “Configuring IPv6 Addressing” section on page 7-8.

Step 6
(Optional) Allow same security level communication, either by allowing communication between two interfaces or by allowing traffic to enter and exit the same interface. See the “Allowing Same Security Level Communication” section on page 7-12.

Configuring General Interface Parameters

This procedure describes how to set the name, security level, IPv4 address and other options.

For the ASASM, you must configure interface parameters for the following interface types:
• VLAN interfaces

Guidelines and Limitations

• If you are using failover, do not use this procedure to name interfaces that you are reserving for failover and Stateful Failover communications. See the “Configuring Active/Standby Failover” section on page 50-7 or the “Configuring Active/Active Failover” section on page 51-8 to configure the failover and state links.

Restrictions

• PPPoE is not supported in multiple context mode.
• PPPoE and DHCP are not supported on the ASASM.

Prerequisites

• Set up your interfaces depending on your model:
  – ASASM—Chapter 2, “Configuring the Switch for Use with the ASA Services Module.”
• In multiple context mode, you can only configure context interfaces that you already assigned to the context in the system configuration according to the “Configuring Multiple Contexts” section on page 6-14.
• In multiple context mode, complete this procedure in the context execution space. To change from the system to a context configuration, enter the change to context name command.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>If you are not already in interface configuration mode, enters interface configuration mode. In multiple context mode, enter the mapped_name if one was assigned using the allocate-interface command.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 2</td>
<td>Names the interface. The name is a text string up to 48 characters, and is not case-sensitive. You can change the name by reentering this command with a new value. Do not enter the no form, because that command causes all commands that refer to that name to be deleted.</td>
</tr>
</tbody>
</table>

Step 3  Do the following:
Completing Interface Configuration in Routed Mode

Example

The following example configures parameters for VLAN 101:

```
hostname(config)# interface vlan 101
hostname(config-if)# nameif inside
hostname(config-if)# security-level 100
hostname(config-if)# ip address 10.1.1.1 255.255.255.0
```

The following example configures parameters in multiple context mode for the context configuration. The interface ID is a mapped name.

```
hostname/contextA(config)# interface int1
hostname/contextA(config-if)# nameif outside
hostname/contextA(config-if)# security-level 100
hostname/contextA(config-if)# ip address 10.1.2.1 255.255.255.0
```

What to Do Next

- (Optional) Configure the MAC address and the MTU. See the “Configuring the MAC Address and MTU” section on page 7-6.
- (Optional) Configure IPv6 addressing. See the “Configuring IPv6 Addressing” section on page 7-8.

Configuring the MAC Address and MTU

This section describes how to configure MAC addresses for interfaces and how to set the MTU.
Information About MAC Addresses

For the ASASM, all VLANs use the same MAC address provided by the backplane.

In multiple context mode, if you share an interface between contexts, you can assign a unique MAC address to the interface in each context. This feature lets the ASASM easily classify packets into the appropriate context. Using a shared interface without unique MAC addresses is possible, but has some limitations. See the “How the ASA Classifies Packets” section on page 6-3 for more information. You can assign each MAC address manually, or you can automatically generate MAC addresses for shared interfaces in contexts. See the “Automatically Assigning MAC Addresses to Context Interfaces” section on page 6-21 to automatically generate MAC addresses. If you automatically generate MAC addresses, you can use this procedure to override the generated address.

For single context mode, or for interfaces that are not shared in multiple context mode, you might want to assign unique MAC addresses to subinterfaces. For example, your service provider might perform access control based on the MAC address.

Information About the MTU

The MTU is the maximum datagram size that is sent on a connection. Data that is larger than the MTU value is fragmented before being sent.

The ASASM supports IP path MTU discovery (as defined in RFC 1191), which allows a host to dynamically discover and cope with the differences in the maximum allowable MTU size of the various links along the path. Sometimes, the ASASM cannot forward a datagram because the packet is larger than the MTU that you set for the interface, but the “don't fragment” (DF) bit is set. The network software sends a message to the sending host, alerting it to the problem. The host has to fragment packets for the destination so that they fit the smallest packet size of all the links along the path.

The default MTU is 1500 bytes in a block for Ethernet interfaces. This value is sufficient for most applications, but you can pick a lower number if network conditions require it.

Jumbo frames are supported by default on the ASASM. A jumbo frame is an Ethernet packet larger than the standard maximum of 1518 bytes (including Layer 2 header and FCS), up to 9216 bytes. Jumbo frames require extra memory to process, and assigning more memory for jumbo frames might limit the maximum use of other features, such as access lists. To use jumbo frames, set the value higher, for example, to 9000 bytes.

Prerequisites

- Set up your interfaces depending on your model:
  - ASASM—Chapter 2, “Configuring the Switch for Use with the ASA Services Module.”
- In multiple context mode, you can only configure context interfaces that you already assigned to the context in the system configuration according to the “Configuring Multiple Contexts” section on page 6-14.
- In multiple context mode, complete this procedure in the context execution space. To change from the system to a context configuration, enter the `changepolicy context name` command.
Completing Interface Configuration in Routed Mode

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>If you are not already in interface configuration mode, enters interface configuration mode. In multiple context mode, enter the mapped_name if one was assigned using the allocate-interface command.</td>
</tr>
<tr>
<td>hostname(config)# interface (vlan number</td>
<td>mapped_name)</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# interface vlan 100</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Assigns a private MAC address to this interface. The mac_address is in H.H.H format, where H is a 16-bit hexadecimal digit. For example, the MAC address 00-0C-F1-42-4C-DE is entered as 000C.F142.4CDE. The first two bytes of a manual MAC address cannot be A2 if you also want to use auto-generated MAC addresses. For use with failover, set the standby MAC address. If the active unit fails over and the standby unit becomes active, the new active unit starts using the active MAC addresses to minimize network disruption, while the old active unit uses the standby address.</td>
</tr>
<tr>
<td>mac-address mac_address [standby mac_address]</td>
<td>hostname(config-if)# mac-address 000C.F142.4CDE</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>Sets the MTU between 300 and 65,535 bytes. The default is 1500 bytes.</td>
</tr>
<tr>
<td>mtu interface_name bytes</td>
<td>hostname(config)# mtu inside 9200</td>
</tr>
</tbody>
</table>

What to Do Next

(Optional) Configure IPv6 addressing. See the “Configuring IPv6 Addressing” section on page 7-8.

Configuring IPv6 Addressing

This section describes how to configure IPv6 addressing. For more information about IPv6, see the “Information About IPv6 Support” section on page 19-9 and the “IPv6 Addresses” section on page B-5. This section includes the following topics:

- Information About IPv6, page 7-8
- Configuring a Global IPv6 Address and Other Options, page 7-10

Information About IPv6

This section includes information about how to configure IPv6, and includes the following topics:

- IPv6 Addressing, page 7-8
- Duplicate Address Detection, page 7-9
- Modified EUI-64 Interface IDs, page 7-9

IPv6 Addressing

You can configure two types of unicast addresses for IPv6:
Global—The global address is a public address that you can use on the public network.

Link-local—The link-local address is a private address that you can only use on the directly-connected network. Routers do not forward packets using link-local addresses; they are only for communication on a particular physical network segment. They can be used for address configuration or for the ND functions such as address resolution and neighbor discovery.

At a minimum, you need to configure a link-local addresses for IPv6 to operate. If you configure a global address, a link-local address is automatically configured on the interface, so you do not also need to specifically configure a link-local address. If you do not configure a global address, then you need to configure the link-local address, either automatically or manually.

**Note**
If you want to only configure the link-local addresses, see the `ipv6 enable` (to auto-configure) or `ipv6 address link-local` (to manually configure) command in the command reference.

### Duplicate Address Detection

During the stateless autoconfiguration process, duplicate address detection (DAD) verifies the uniqueness of new unicast IPv6 addresses before the addresses are assigned to interfaces (the new addresses remain in a tentative state while duplicate address detection is performed). Duplicate address detection is performed first on the new link-local address. When the link-local address is verified as unique, then duplicate address detection is performed all the other IPv6 unicast addresses on the interface.

Duplicate address detection is suspended on interfaces that are administratively down. While an interface is administratively down, the unicast IPv6 addresses assigned to the interface are set to a pending state. An interface returning to an administratively up state restarts duplicate address detection for all of the unicast IPv6 addresses on the interface.

When a duplicate address is identified, the state of the address is set to DUPLICATE, the address is not used, and the following error message is generated:

```
%ASA-4-325002: Duplicate address ipv6_address/MAC_address on interface
```

If the duplicate address is the link-local address of the interface, the processing of IPv6 packets is disabled on the interface. If the duplicate address is a global address, the address is not used. However, all configuration commands associated with the duplicate address remain as configured while the state of the address is set to DUPLICATE.

If the link-local address for an interface changes, duplicate address detection is performed on the new link-local address and all of the other IPv6 address associated with the interface are regenerated (duplicate address detection is performed only on the new link-local address).

The ASASM uses neighbor solicitation messages to perform duplicate address detection. By default, the number of times an interface performs duplicate address detection is 1.

### Modified EUI-64 Interface IDs

RFC 3513: Internet Protocol Version 6 (IPv6) Addressing Architecture requires that the interface identifier portion of all unicast IPv6 addresses, except those that start with binary value 000, be 64 bits long and be constructed in Modified EUI-64 format. The ASASM can enforce this requirement for hosts attached to the local link.

When this feature is enabled on an interface, the source addresses of IPv6 packets received on that interface are verified against the source MAC addresses to ensure that the interface identifiers use the Modified EUI-64 format. If the IPv6 packets do not use the Modified EUI-64 format for the interface identifier, the packets are dropped and the following system log message is generated:
Completing Interface Configuration in Routed Mode

%ASA-3-325003: EUI-64 source address check failed.

The address format verification is only performed when a flow is created. Packets from an existing flow are not checked. Additionally, the address verification can only be performed for hosts on the local link. Packets received from hosts behind a router will fail the address format verification, and be dropped, because their source MAC address will be the router MAC address and not the host MAC address.

Configuring a Global IPv6 Address and Other Options

To configure a global IPv6 address and other options, perform the following steps.

Note

Configuring the global address automatically configures the link-local address, so you do not need to configure it separately.

Restrictions

The ASASM does not support IPv6 anycast addresses.

Prerequisites

- Set up your interfaces depending on your model:
  - ASASM—Chapter 2, “Configuring the Switch for Use with the ASA Services Module.”
- In multiple context mode, you can only configure context interfaces that you already assigned to the context in the system configuration according to the “Configuring Multiple Contexts” section on page 6-14.
- In multiple context mode, complete this procedure in the context execution space. To change from the system to a context configuration, enter the `change to context name` command.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1**
hostname(config)# interface {vlan number | mapped_name} | If you are not already in interface configuration mode, enters interface configuration mode.
In multiple context mode, enter the `mapped_name` if one was assigned using the `allocate-interface` command. |

Example:

hostname(config)# interface gigabitethernet 0/0

| Step 2 | Do one of the following: |
### Command

**ipv6 address autoconfig**

**Example:**
```plaintext
hostname(config-if)# ipv6 address autoconfig
```

**Purpose:** Enables stateless autoconfiguration on the interface. Enabling stateless autoconfiguration on the interface configures IPv6 addresses based on prefixes received in Router Advertisement messages. A link-local address, based on the Modified EUI-64 interface ID, is automatically generated for the interface when stateless autoconfiguration is enabled.

**Note** Although RFC 4862 specifies that hosts configured for stateless autoconfiguration do not send Router Advertisement messages, the ASASM does send Router Advertisement messages in this case. See the **ipv6 nd suppress-ra** command to suppress messages.

**ipv6 address ipv6-address/prefix-length [standby ipv6-address]**

**Example:**
```plaintext
hostname(config-if)# ipv6 address 2001:0DB8::BA98:0:3210/48
```

**Purpose:** Assigns a global address to the interface. When you assign a global address, the link-local address is automatically created for the interface.

**standby** specifies the interface address used by the secondary unit or failover group in a failover pair.

See the “IPv6 Addresses” section on page B-5 for more information about IPv6 addressing.

**ipv6 address ipv6-prefix/prefix-length eui-64**

**Example:**
```plaintext
hostname(config-if)# ipv6 address 2001:0DB8::BA98::/48 eui-64
```

**Purpose:** Assigns a global address to the interface by combining the specified prefix with an interface ID generated from the interface MAC address using the Modified EUI-64 format. When you assign a global address, the link-local address is automatically created for the interface.

**You do not need to specify the standby address; the interface ID will be generated automatically.**

See the “IPv6 Addresses” section on page B-5 for more information about IPv6 addressing.

**Step 3 (Optional)**

**ipv6 nd suppress-ra**

**Example:**
```plaintext
hostname(config-if)# ipv6 nd suppress-ra
```

**Purpose:** Suppresses Router Advertisement messages on an interface. By default, Router Advertisement messages are automatically sent in response to router solicitation messages. You may want to disable these messages on any interface for which you do not want the ASASM to supply the IPv6 prefix (for example, the outside interface).

**Step 4 (Optional)**

**ipv6 nd dad attempts value**

**Example:**
```plaintext
hostname(config-if)# ipv6 nd dad attempts 3
```

**Purpose:** Changes the number of duplicate address detection attempts. The value argument can be any value from 0 to 600. Setting the value argument to 0 disables duplicate address detection on the interface.

By default, the number of times an interface performs duplicate address detection is 1. See the “Duplicate Address Detection” section on page 7-9 for more information.
Completing Interface Configuration in Routed Mode

Chapter 7  Configuring Interfaces (Routed Mode)

Allowing Same Security Level Communication

By default, interfaces on the same security level cannot communicate with each other, and packets cannot enter and exit the same interface. This section describes how to enable inter-interface communication when interfaces are on the same security level, and how to enable intra-interface communication.

Information About Inter-Interface Communication

Allowing interfaces on the same security level to communicate with each other provides the following benefits:

- You can configure more than 101 communicating interfaces.
  
  If you use different levels for each interface and do not assign any interfaces to the same security level, you can configure only one interface per level (0 to 100).

- You want traffic to flow freely between all same security interfaces without access lists.

If you enable same security interface communication, you can still configure interfaces at different security levels as usual.

Information About Intra-Interface Communication

All traffic allowed by this feature is still subject to firewall rules. Be careful not to create an asymmetric routing situation that can cause return traffic not to traverse the ASASM.

For the ASASM, before you can enable this feature, you must first correctly configure the MSFC so that packets are sent to the ASASM MAC address instead of being sent directly through the switch to the destination host. Figure 7-1 shows a network where hosts on the same interface need to communicate.
The following sample configuration shows the Cisco IOS `route-map` commands used to enable policy routing in the network shown in Figure 7-1:

```plaintext
route-map intra-inter3 permit 0
  match ip address 103
  set interface Vlan20
  set ip next-hop 10.6.34.7

route-map intra-inter2 permit 20
  match ip address 102
  set interface Vlan20
  set ip next-hop 10.6.34.7

route-map intra-inter1 permit 10
  match ip address 101
  set interface Vlan20
  set ip next-hop 10.6.34.7
```

### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>same-security-traffic permit</code></td>
<td>Enables interfaces on the same security level so that they can communicate with each other.</td>
</tr>
<tr>
<td><code>intra-interface</code></td>
<td></td>
</tr>
<tr>
<td><code>same-security-traffic permit</code></td>
<td>Enables communication between hosts connected to the same interface.</td>
</tr>
<tr>
<td><code>intra-interface</code></td>
<td></td>
</tr>
</tbody>
</table>
Turning Off and Turning On Interfaces

This section describes how to turn off and on an interface on the ASASM.

All interfaces are enabled by default. In multiple context mode, if you disable or reenable the interface within a context, only that context interface is affected. But if you disable or reenable the interface in the system execution space, then you affect that interface for all contexts.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>If you are not already in interface configuration mode, enters interface configuration mode.</td>
</tr>
<tr>
<td>hostname(config)# interface (vlan number</td>
<td>In multiple context mode, enter the mapped_name if one was assigned using the allocate-interface command.</td>
</tr>
<tr>
<td>mapped_name)</td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# interface vlan 100</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Disables the interface.</td>
</tr>
<tr>
<td>shutdown</td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config-if)# shutdown</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>Reenables the interface.</td>
</tr>
<tr>
<td>no shutdown</td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config-if)# no shutdown</td>
<td></td>
</tr>
</tbody>
</table>

Monitoring Interfaces

To monitor interfaces, enter one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show interface</td>
<td>Displays interface statistics.</td>
</tr>
<tr>
<td>show interface ip brief</td>
<td>Displays interface IP addresses and status.</td>
</tr>
</tbody>
</table>
Feature History for Interfaces in Routed Mode

Table 7-1 lists the release history for this feature.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Increased VLANs</td>
<td>7.0(5)</td>
<td>Increased the following limits:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ASA5510 Base license VLANs from 0 to 10.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ASA5510 Security Plus license VLANs from 10 to 25.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ASA5520 VLANs from 25 to 100.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ASA5540 VLANs from 100 to 200.</td>
</tr>
<tr>
<td>Increased VLANs</td>
<td>7.2(2)</td>
<td>The maximum number of VLANs for the Security Plus license on the ASA 5505 was increased from 5 (3 fully functional; 1 failover; one restricted to a backup interface) to 20 fully functional interfaces. In addition, the number of trunk ports was increased from 1 to 8. Now there are 20 fully functional interfaces, you do not need to use the backup interface command to cripple a backup ISP interface; you can use a fully-functional interface for it. The backup interface command is still useful for an Easy VPN configuration. VLAN limits were also increased for the ASA 5510 (from 10 to 50 for the Base license, and from 25 to 100 for the Security Plus license), the ASA 5520 (from 100 to 150), the ASA 5550 (from 200 to 250).</td>
</tr>
<tr>
<td>Gigabit Ethernet Support for the ASA 5510 Security Plus License</td>
<td>7.2(3)</td>
<td>The ASA 5510 now supports GE (Gigabit Ethernet) for port 0 and 1 with the Security Plus license. If you upgrade the license from Base to Security Plus, the capacity of the external Ethernet0/0 and Ethernet0/1 ports increases from the original FE (Fast Ethernet) (100 Mbps) to GE (1000 Mbps). The interface names will remain Ethernet 0/0 and Ethernet 0/1. Use the speed command to change the speed on the interface and use the show interface command to see what speed is currently configured for each interface.</td>
</tr>
<tr>
<td>Native VLAN support for the ASA 5505</td>
<td>7.2(4)/8.0(4)</td>
<td>You can now include the native VLAN in an ASA 5505 trunk port. We introduced the following command: switchport trunk native vlan.</td>
</tr>
</tbody>
</table>
Jumbo packet support for the ASA 5580 8.1(1) The Cisco ASA 5580 supports jumbo frames. A jumbo frame is an Ethernet packet larger than the standard maximum of 1518 bytes (including Layer 2 header and FCS), up to 9216 bytes. You can enable support for jumbo frames for all interfaces by increasing the amount of memory to process Ethernet frames. Assigning more memory for jumbo frames might limit the maximum use of other features, such as access lists.

We introduced the following command: `jumbo-frame reservation`.

Increased VLANs for the ASA 5580 8.1(2) The number of VLANs supported on the ASA 5580 are increased from 100 to 250.

IPv6 support for transparent mode 8.2(1) IPv6 support was introduced for transparent firewall mode.

Support for Pause Frames for Flow Control on the ASA 5580 10 Gigabit Ethernet Interfaces 8.2(2) You can now enable pause (XOFF) frames for flow control. We introduced the following command: `flowcontrol`.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jumbo packet support for the ASA 5580</td>
<td>8.1(1)</td>
<td>The Cisco ASA 5580 supports jumbo frames. A jumbo frame</td>
</tr>
<tr>
<td></td>
<td></td>
<td>is an Ethernet packet larger than the standard maximum</td>
</tr>
<tr>
<td></td>
<td></td>
<td>of 1518 bytes (including Layer 2 header and FCS), up to</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9216 bytes. You can enable support for jumbo frames for</td>
</tr>
<tr>
<td></td>
<td></td>
<td>all interfaces by increasing the amount of memory to</td>
</tr>
<tr>
<td></td>
<td></td>
<td>process Ethernet frames. Assigning more memory for</td>
</tr>
<tr>
<td></td>
<td></td>
<td>jumbo frames might limit the maximum use of other</td>
</tr>
<tr>
<td></td>
<td></td>
<td>features, such as access lists. We introduced the</td>
</tr>
<tr>
<td></td>
<td></td>
<td>following command: <code>jumbo-frame reservation</code>.</td>
</tr>
<tr>
<td>Increased VLANs for the ASA 5580</td>
<td>8.1(2)</td>
<td>The number of VLANs supported on the ASA 5580 are</td>
</tr>
<tr>
<td></td>
<td></td>
<td>increased from 100 to 250.</td>
</tr>
<tr>
<td>IPv6 support for transparent mode</td>
<td>8.2(1)</td>
<td>IPv6 support was introduced for transparent firewall</td>
</tr>
<tr>
<td></td>
<td></td>
<td>mode.</td>
</tr>
<tr>
<td>Support for Pause Frames for Flow Control on</td>
<td>8.2(2)</td>
<td>You can now enable pause (XOFF) frames for flow control.</td>
</tr>
<tr>
<td>the ASA 5580 10 Gigabit Ethernet Interfaces</td>
<td></td>
<td>We introduced the following command: <code>flowcontrol</code>.</td>
</tr>
</tbody>
</table>
CHAPTER 8

Configuring Interfaces (Transparent Mode)

This chapter includes tasks to complete the interface configuration in transparent firewall mode. This chapter includes the following sections:

- Information About Completing Interface Configuration in Transparent Mode, page 8-1
- Licensing Requirements for Completing Interface Configuration in Transparent Mode, page 8-2
- Guidelines and Limitations, page 8-3
- Default Settings, page 8-4
- Completing Interface Configuration in Transparent Mode, page 8-5
- Turning Off and Turning On Interfaces, page 8-13
- Monitoring Interfaces, page 8-14
- Configuration Examples for Interfaces in Transparent Mode, page 8-14
- Feature History for Interfaces in Transparent Mode, page 8-15

Note

For multiple context mode, complete the tasks in this section in the context execution space. Enter the `changeto context name` command to change to the context you want to configure.

Information About Completing Interface Configuration in Transparent Mode

This section includes the following topics:

- Bridge Groups in Transparent Mode, page 8-1
- Security Levels, page 8-2

Bridge Groups in Transparent Mode

If you do not want the overhead of security contexts, or want to maximize your use of security contexts, you can group interfaces together in a bridge group, and then configure multiple bridge groups, one for each network. Bridge group traffic is isolated from other bridge groups; traffic is not routed to another bridge group within the ASASM, and traffic must exit the ASASM before it is routed by an external router back to another bridge group in the ASASM. Although the bridging functions are separate for each
bridge group, many other functions are shared between all bridge groups. For example, all bridge groups share a syslog server or AAA server configuration. For complete security policy separation, use security contexts with one bridge group in each context. At least one bridge group is required per context or in single mode.

Each bridge group requires a management IP address.

**Note**

The ASASM does not support traffic on secondary networks; only traffic on the same network as the management IP address is supported.

### Security Levels

Each interface must have a security level from 0 (lowest) to 100 (highest). For example, you should assign your most secure network, such as the inside host network, to level 100. While the outside network connected to the Internet can be level 0. Other networks, such as DMZs can be in between. You can assign interfaces to the same security level. See the “Allowing Same Security Level Communication” section on page 8-13 for more information.

The level controls the following behavior:

- **Network access**—By default, there is an implicit permit from a higher security interface to a lower security interface (outbound). Hosts on the higher security interface can access any host on a lower security interface. You can limit access by applying an access list to the interface.

  If you enable communication for same security interfaces (see the “Allowing Same Security Level Communication” section on page 8-13), there is an implicit permit for interfaces to access other interfaces on the same security level or lower.

- **Inspection engines**—Some application inspection engines are dependent on the security level. For same security interfaces, inspection engines apply to traffic in either direction.
  - NetBIOS inspection engine—Applied only for outbound connections.
  - SQL*Net inspection engine—If a control connection for the SQL*Net (formerly OraServ) port exists between a pair of hosts, then only an inbound data connection is permitted through the ASASM.

- **Filtering**—HTTP(S) and FTP filtering applies only for outbound connections (from a higher level to a lower level).

  If you enable communication for same security interfaces, you can filter traffic in either direction.

- **established** command—This command allows return connections from a lower security host to a higher security host if there is already an established connection from the higher level host to the lower level host.

  If you enable communication for same security interfaces, you can configure established commands for both directions.

### Licensing Requirements for Completing Interface Configuration in Transparent Mode
Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

Context Mode Guidelines
- For the ASASM in multiple context mode, configure switch ports and VLANs on the switch, and then assign VLANs to the ASASM according to Chapter 2, “Configuring the Switch for Use with the ASA Services Module.”
- You can only configure context interfaces that you already assigned to the context in the system configuration using the `allocate-interface` command.

Firewall Mode Guidelines
- You can configure up to 8 bridge groups in single mode or per context in multiple mode. Note that you must use at least 1 bridge group; data interfaces must belong to a bridge group.

  Note  Although you can configure multiple bridge groups on the ASA 5505, the restriction of 2 data interfaces in transparent mode on the ASA 5505 means you can only effectively use 1 bridge group.

- Each bridge group can include up to 4 interfaces.
- For IPv4, a management IP address is required for each bridge group for both management traffic and for traffic to pass through the ASASM.

Unlike routed mode, which requires an IP address for each interface, a transparent firewall has an IP address assigned to the entire bridge group. The ASASM uses this IP address as the source address for packets originating on the ASASM, such as system messages or AAA communications.

The management IP address must be on the same subnet as the connected network. You cannot set the subnet to a host subnet (255.255.255.255). The ASASM does not support traffic on secondary networks; only traffic on the same network as the management IP address is supported. See the “Configuring Bridge Groups” section on page 8-5 for more information about management IP subnets.

- For IPv6, at a minimum you need to configure link-local addresses for each interface for through traffic. For full functionality, including the ability to manage the ASASM, you need to configure a global IPv6 address for each bridge group.
- For multiple context mode, each context must use different interfaces; you cannot share an interface across contexts.
- For multiple context mode, each context typically uses a different subnet. You can use overlapping subnets, but your network topology requires router and NAT configuration to make it possible from a routing standpoint.

### Model License Requirement

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASASM</td>
<td>VLANs: Base License: 1000</td>
</tr>
</tbody>
</table>
Failover Guidelines
Do not finish configuring failover interfaces with the procedures in this chapter. See the “Configuring Active/Standby Failover” section on page 50-7 or the “Configuring Active/Active Failover” section on page 51-8 to configure the failover and state links. In multiple context mode, failover interfaces are configured in the system configuration.

IPv6 Guidelines
- Supports IPv6.
- No support for IPv6 anycast addresses in transparent mode.

VLAN ID Guidelines for the ASASM
You can add any VLAN ID to the configuration, but only VLANs that are assigned to the ASASM by the switch can pass traffic. To view all VLANs assigned to the ASASM, use the `show vlan` command.

If you add an interface for a VLAN that is not yet assigned to the ASASM by the switch, the interface will be in the down state. When you assign the VLAN to the ASASM, the interface changes to an up state. See the `show interface` command for more information about interface states.

Default Settings

This section lists default settings for interfaces if you do not have a factory default configuration. For information about the factory default configurations, see the “Working with the Configuration” section on page 3-11.

Default Security Level
The default security level is 0. If you name an interface “inside” and you do not set the security level explicitly, then the ASASM sets the security level to 100.

Note
If you change the security level of an interface, and you do not want to wait for existing connections to time out before the new security information is used, you can clear the connections using the `clear local-host` command.

Default State of Interfaces for the ASASM
- In single mode or in the system execution space, VLAN interfaces are enabled by default.
- In multiple context mode, all allocated interfaces are enabled by default, no matter what the state of the interface is in the system execution space. However, for traffic to pass through the interface, the interface also has to be enabled in the system execution space. If you shut down an interface in the system execution space, then that interface is down in all contexts that share it.

Jumbo Frame Support
By default, the ASASM supports jumbo frames. Just configure the MTU for the desired packet size according to the “Configuring the MAC Address and MTU” section on page 8-8.
Completing Interface Configuration in Transparent Mode

This section includes the following topics:

- Task Flow for Completing Interface Configuration, page 8-5
- Configuring Bridge Groups, page 8-5
- Configuring General Interface Parameters, page 8-6
- Configuring the MAC Address and MTU, page 8-8
- Configuring IPv6 Addressing, page 8-9
- Allowing Same Security Level Communication, page 8-13

Task Flow for Completing Interface Configuration

Step 1 Set up your interfaces depending on your model:
- ASASM—Chapter 2, “Configuring the Switch for Use with the ASA Services Module.”

Step 2 (Multiple context mode) Allocate interfaces to the context according to the “Configuring Multiple Contexts” section on page 6-14.

Step 3 (Multiple context mode) Enter the chgneto context name command to change to the context you want to configure. Configure one or more bridge groups, including the IPv4 address. See the “Configuring Bridge Groups” section on page 8-5.

Step 4 Configure general interface parameters, including the interface name and security level. See the “Configuring General Interface Parameters” section on page 8-6.

Step 5 (Optional) Configure the MAC address and the MTU. See the “Configuring the MAC Address and MTU” section on page 8-8.

Step 6 (Optional) Configure IPv6 addressing. See the “Configuring IPv6 Addressing” section on page 8-9.

Step 7 (Optional) Allow same security level communication, either by allowing communication between two interfaces or by allowing traffic to enter and exit the same interface. See the “Allowing Same Security Level Communication” section on page 8-13.

Configuring Bridge Groups

Each bridge group requires a management IP address. The ASASM uses this IP address as the source address for packets originating from the bridge group. The management IP address must be on the same subnet as the connected network. For IPv4 traffic, the management IP address is required to pass any traffic. For IPv6 traffic, you must, at a minimum, configure the link-local addresses to pass traffic, but a global management address is recommended for full functionality, including remote management and other management operations.

Guidelines and Limitations

You can configure up to 8 bridge groups in single mode or per context in multiple mode. Note that you must use at least one bridge group; data interfaces must belong to a bridge group.
Completing Interface Configuration in Transparent Mode

Note
For a separate management interface (for supported models), a non-configurable bridge group (ID 101) is automatically added to your configuration. This bridge group is not included in the bridge group limit.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td>interface bvi bridge_group_number</td>
<td>Creates a bridge group, where <em>bridge_group_number</em> is an integer between 1 and 100.</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td>ip address <em>ip_address</em> [mask] [standby <em>ip_address</em>]</td>
<td>Specifies the management IP address for the bridge group. Do not assign a host address (/32 or 255.255.255.255) to the bridge group. Also, do not use other subnets that contain fewer than 3 host addresses (one each for the upstream router, downstream router, and transparent firewall) such as a /30 subnet (255.255.255.252). The ASASM drops all ARP packets to or from the first and last addresses in a subnet. Therefore, if you use a /30 subnet and assign a reserved address from that subnet to the upstream router, then the ASASM drops the ARP request from the downstream router to the upstream router. The ASASM does not support traffic on secondary networks; only traffic on the same network as the management IP address is supported. The <em>standby</em> keyword and address is used for failover.</td>
</tr>
</tbody>
</table>

Example:
hostname(config)# interface bvi 1
hostname(config-if)# ip address 10.1.3.1 255.255.255.0 standby 10.1.3.2

Examples
The following example sets the management address and standby address of bridge group 1:
hostname(config)# interface bvi 1
hostname(config-if)# ip address 10.1.3.1 255.255.255.0 standby 10.1.3.2

What to Do Next
Configure general interface parameters. See the “Configuring General Interface Parameters” section on page 8-6.

Configuring General Interface Parameters

This procedure describes how to set the name, security level, and bridge group for each transparent interface.

To configure a separate management interface, see the “” section on page 8-8.

For the ASASM, you must configure interface parameters for the following interface types:

- VLAN interfaces
Completing Interface Configuration in Transparent Mode

Guidelines and Limitations

- You can configure up to four interfaces per bridge group.
- For information about security levels, see the “Security Levels” section on page 8-2.
- If you are using failover, do not use this procedure to name interfaces that you are reserving for failover and Stateful Failover communications. See the “Configuring Active/Standby Failover” section on page 50-7 or the “Configuring Active/Active Failover” section on page 51-8 to configure the failover and state links.

Prerequisites

- Set up your interfaces depending on your model:
  - ASASM—Chapter 2, “Configuring the Switch for Use with the ASA Services Module.”
- In multiple context mode, you can only configure context interfaces that you already assigned to the context in the system configuration according to the “Configuring Multiple Contexts” section on page 6-14.
- In multiple context mode, complete this procedure in the context execution space. To change from the system to a context configuration, enter the `changeto context name` command.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>hostname(config)# interface {vlan number</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# interface vlan 100 In multiple context mode, enter the mapped_name if one was assigned using the allocate-interface command.</td>
</tr>
<tr>
<td>Step 2</td>
<td>bridge-group number Assigns the interface to a bridge group, where number is an integer between 1 and 100. You can assign up to four interfaces to a bridge group. You cannot assign the same interface to more than one bridge group.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config-if)# bridge-group 1</td>
</tr>
<tr>
<td>Step 3</td>
<td>nameif name Names the interface. The name is a text string up to 48 characters, and is not case-sensitive. You can change the name by reentering this command with a new value. Do not enter the no form, because that command causes all commands that refer to that name to be deleted.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config-if)# nameif inside</td>
</tr>
<tr>
<td>Step 4</td>
<td>security-level number Sets the security level, where number is an integer between 0 (lowest) and 100 (highest).</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config-if)# security-level 50</td>
</tr>
</tbody>
</table>

What to Do Next

- (Optional) Configure the MAC address and the MTU. See the “Configuring the MAC Address and MTU” section on page 8-8.
- (Optional) Configure IPv6 addressing. See the “Configuring IPv6 Addressing” section on page 8-9.
Configuring the MAC Address and MTU

This section describes how to configure MAC addresses for interfaces and how to set the MTU.

Information About MAC Addresses

For the ASASM, all VLANs use the same MAC address provided by the backplane.

In multiple context mode, if you share an interface between contexts, you can assign a unique MAC address to the interface in each context. This feature lets the ASASM easily classify packets into the appropriate context. Using a shared interface without unique MAC addresses is possible, but has some limitations. See the “How the ASA Classifies Packets” section on page 6-3 for more information. You can assign each MAC address manually, or you can automatically generate MAC addresses for shared interfaces in contexts. See the “Automatically Assigning MAC Addresses to Context Interfaces” section on page 6-21 to automatically generate MAC addresses. If you automatically generate MAC addresses, you can use this procedure to override the generated address.

For single context mode, or for interfaces that are not shared in multiple context mode, you might want to assign unique MAC addresses to subinterfaces. For example, your service provider might perform access control based on the MAC address.

Information About the MTU

The MTU is the maximum datagram size that is sent on a connection. Data that is larger than the MTU value is fragmented before being sent.

The ASASM supports IP path MTU discovery (as defined in RFC 1191), which allows a host to dynamically discover and cope with the differences in the maximum allowable MTU size of the various links along the path. Sometimes, the ASASM cannot forward a datagram because the packet is larger than the MTU that you set for the interface, but the “don't fragment” (DF) bit is set. The network software sends a message to the sending host, alerting it to the problem. The host has to fragment packets for the destination so that they fit the smallest packet size of all the links along the path.

The default MTU is 1500 bytes in a block for Ethernet interfaces. This value is sufficient for most applications, but you can pick a lower number if network conditions require it.

Jumbo frames are supported by default on the ASASM. A jumbo frame is an Ethernet packet larger than the standard maximum of 1518 bytes (including Layer 2 header and FCS), up to 9216 bytes. Jumbo frames require extra memory to process, and assigning more memory for jumbo frames might limit the maximum use of other features, such as access lists. To use jumbo frames, set the value higher, for example, to 9000 bytes.

Prerequisites

- Set up your interfaces depending on your model:
  - ASASM—Chapter 2, “Configuring the Switch for Use with the ASA Services Module.”
- In multiple context mode, you can only configure context interfaces that you already assigned to the context in the system configuration according to the “Configuring Multiple Contexts” section on page 6-14.
- In multiple context mode, complete this procedure in the context execution space. To change from the system to a context configuration, enter the `changectx context name` command.
Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>If you are not already in interface configuration mode, enters interface configuration mode.</td>
</tr>
<tr>
<td>hostname(config)# interface (vlan number</td>
<td>In multiple context mode, enter the mapped_name if one was assigned using the allocate-interface command.</td>
</tr>
<tr>
<td></td>
<td>mapped_name)</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# interface vlan 100</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Assigns a private MAC address to this interface. The mac_address is in H.H.H format, where H is a 16-bit hexadecimal digit. For example, the MAC address 00-0C-F1-42-4C-DE is entered as 000C.F142.4CDE.</td>
</tr>
<tr>
<td>mac-address mac_address</td>
<td>The first two bytes of a manual MAC address cannot be A2 if you also want to use auto-generated MAC addresses.</td>
</tr>
<tr>
<td>[standby mac_address]</td>
<td>For use with failover, set the standby MAC address. If the active unit fails over and the standby unit becomes active, the new active unit starts using the active MAC addresses to minimize network disruption, while the old active unit uses the standby address.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config-if)# mac-address 000C.F142.4CDE</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>Sets the MTU between 300 and 65,535 bytes. The default is 1500 bytes.</td>
</tr>
<tr>
<td>mtu interface_name bytes</td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# mtu inside 9200</td>
<td></td>
</tr>
</tbody>
</table>

What to Do Next

(Optional) Configure IPv6 addressing. See the “Configuring IPv6 Addressing” section on page 8-9.

Configuring IPv6 Addressing

This section describes how to configure IPv6 addressing. For more information about IPv6, see the “Information About IPv6 Support” section on page 19-9 and the “IPv6 Addresses” section on page B-5.

This section includes the following topics:

- Information About IPv6, page 8-9
- Configuring a Global IPv6 Address and Other Options, page 8-11

Information About IPv6

This section includes information about how to configure IPv6, and includes the following topics:

- IPv6 Addressing, page 8-10
- Duplicate Address Detection, page 8-10
- Modified EUI-64 Interface IDs, page 8-11
- Unsupported Commands, page 8-11
IPv6 Addressing

You can configure two types of unicast addresses for IPv6:

- **Global**—The global address is a public address that you can use on the public network. This address needs to be configured for each bridge group, and not per-interface. You can also configure a global IPv6 address for the management interface.

- **Link-local**—The link-local address is a private address that you can only use on the directly-connected network. Routers do not forward packets using link-local addresses; they are only for communication on a particular physical network segment. They can be used for address configuration or for the ND functions such as address resolution and neighbor discovery. Because the link-local address is only available on a segment, and is tied to the interface MAC address, you need to configure the link-local address per interface.

At a minimum, you need to configure a link-local address for IPv6 to operate. If you configure a global address, a link-local addresses is automatically configured on each interface, so you do not also need to specifically configure a link-local address. If you do not configure a global address, then you need to configure the link-local address, either automatically or manually.

**Note**
If you want to only configure the link-local addresses, see the `ipv6 enable` (to auto-configure) or `ipv6 address link-local` (to manually configure) command in the command reference.

Duplicate Address Detection

During the stateless autoconfiguration process, duplicate address detection (DAD) verifies the uniqueness of new unicast IPv6 addresses before the addresses are assigned to interfaces (the new addresses remain in a tentative state while duplicate address detection is performed). Duplicate address detection is performed first on the new link-local address. When the link local address is verified as unique, then duplicate address detection is performed all the other IPv6 unicast addresses on the interface.

Duplicate address detection is suspended on interfaces that are administratively down. While an interface is administratively down, the unicast IPv6 addresses assigned to the interface are set to a pending state. An interface returning to an administratively up state restarts duplicate address detection for all of the unicast IPv6 addresses on the interface.

When a duplicate address is identified, the state of the address is set to DUPLICATE, the address is not used, and the following error message is generated:

```
%ASA-4-325002: Duplicate address ipv6_address/MAC_address on interface
```

If the duplicate address is the link-local address of the interface, the processing of IPv6 packets is disabled on the interface. If the duplicate address is a global address, the address is not used. However, all configuration commands associated with the duplicate address remain as configured while the state of the address is set to DUPLICATE.

If the link-local address for an interface changes, duplicate address detection is performed on the new link-local address and all of the other IPv6 address associated with the interface are regenerated (duplicate address detection is performed only on the new link-local address).

The ASASM uses neighbor solicitation messages to perform duplicate address detection. By default, the number of times an interface performs duplicate address detection is 1.
Modified EUI-64 Interface IDs

RFC 3513: Internet Protocol Version 6 (IPv6) Addressing Architecture requires that the interface identifier portion of all unicast IPv6 addresses, except those that start with binary value 000, be 64 bits long and be constructed in Modified EUI-64 format. The ASASM can enforce this requirement for hosts attached to the local link.

When this feature is enabled on an interface, the source addresses of IPv6 packets received on that interface are verified against the source MAC addresses to ensure that the interface identifiers use the Modified EUI-64 format. If the IPv6 packets do not use the Modified EUI-64 format for the interface identifier, the packets are dropped and the following system log message is generated:

%ASA-3-325003: EUI-64 source address check failed.

The address format verification is only performed when a flow is created. Packets from an existing flow are not checked. Additionally, the address verification can only be performed for hosts on the local link. Packets received from hosts behind a router will fail the address format verification, and be dropped, because their source MAC address will be the router MAC address and not the host MAC address.

Unsupported Commands

The following IPv6 commands are not supported in transparent firewall mode, because they require router capabilities:

- ipv6 address autoconfig
- ipv6 nd prefix
- ipv6 nd ra-interval
- ipv6 nd ra-lifetime
- ipv6 nd suppress-ra

The ipv6 local pool VPN command is not supported, because transparent mode does not support VPN.

Configuring a Global IPv6 Address and Other Options

To configure a global IPv6 address and other options for a bridge group or management interface, perform the following steps.

**Note**

Configuring the global address automatically configures the link-local address, so you do not need to configure it separately.

Restrictions

The ASASM does not support IPv6 anycast addresses.

Prerequisites

- Set up your interfaces depending on your model:
  - ASASM—Chapter 2, “Configuring the Switch for Use with the ASA Services Module.”
- In multiple context mode, you can only configure context interfaces that you already assigned to the context in the system configuration according to the “Configuring Multiple Contexts” section on page 6-14.
Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><strong>interface bvi bridge_group_id</strong>&lt;br&gt;Example:&lt;br&gt;hostname(config)# interface bvi 1</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><strong>ipv6 address ipv6-address/prefix-length [standby ipv6-address]</strong>&lt;br&gt;Example:&lt;br&gt;hostname(config-if)# ipv6 address 2001:0DB8::BA98:0:3210/48</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>(Optional) <strong>ipv6 nd suppress-ra</strong>&lt;br&gt;Example:&lt;br&gt;hostname(config-if)# ipv6 nd suppress-ra</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>(Optional) <strong>ipv6 nd dad attempts value</strong>&lt;br&gt;Example:&lt;br&gt;hostname(config-if)# ipv6 nd dad attempts 3</td>
</tr>
</tbody>
</table>
This section describes how to turn off and on an interface on the ASASM.

All interfaces are enabled by default. In multiple context mode, if you disable or reenable the interface within a context, only that context interface is affected. But if you disable or reenable the interface in the system execution space, then you affect that interface for all contexts.
Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1**

hostname(config)# interface {vlan number | mapped_name}

- If you are not already in interface configuration mode, enters interface configuration mode.
- In multiple context mode, enter the mapped_name if one was assigned using the allocate-interface command.

**Example:**

hostname(config)# interface vlan 100

**Step 2**

shutdown

- Disables the interface.

**Example:**

hostname(config-if)# shutdown

**Step 3**

no shutdown

- Reenables the interface.

**Example:**

hostname(config-if)# no shutdown

---

Monitoring Interfaces

To monitor interfaces, enter one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show interface</td>
<td>Displays interface statistics.</td>
</tr>
<tr>
<td>show interface ip brief</td>
<td>Displays interface IP addresses and status.</td>
</tr>
<tr>
<td>show bridge-group</td>
<td>Shows bridge group information.</td>
</tr>
</tbody>
</table>

---

Configuration Examples for Interfaces in Transparent Mode

The following example includes two bridge groups of three interfaces each, plus a management-only interface:

```plaintext
interface gigabitethernet 0/0
    nameif inside1
    security-level 100
    bridge-group 1
    no shutdown
interface gigabitethernet 0/1
    nameif outside1
    security-level 0
    bridge-group 1
    no shutdown
interface gigabitethernet 0/2
    nameif dmz1
    security-level 50
    bridge-group 1
    no shutdown
interface bvi 1
    ip address 10.1.3.1 255.255.255.0 standby 10.1.3.2
```
Feature History for Interfaces in Transparent Mode

Table 8-1 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Increased VLANs</td>
<td>7.0(5)</td>
<td>Increased the following limits:&lt;br&gt;• ASA5510 Base license VLANs from 0 to 10.&lt;br&gt;• ASA5510 Security Plus license VLANs from 10 to 25.&lt;br&gt;• ASA5520 VLANs from 25 to 100.&lt;br&gt;• ASA5540 VLANs from 100 to 200.</td>
</tr>
<tr>
<td>Increased VLANs</td>
<td>7.2(2)</td>
<td>The maximum number of VLANs for the Security Plus license on the ASA 5505 was increased from 5 (3 fully functional; 1 failover; one restricted to a backup interface) to 20 fully functional interfaces. In addition, the number of trunk ports was increased from 1 to 8. Now there are 20 fully functional interfaces, you do not need to use the backup interface command to cripple a backup ISP interface; you can use a fully-functional interface for it. The backup interface command is still useful for an Easy VPN configuration. VLAN limits were also increased for the ASA 5510 (from 10 to 50 for the Base license, and from 25 to 100 for the Security Plus license), the ASA 5520 (from 100 to 150), the ASA 5550 (from 200 to 250).</td>
</tr>
</tbody>
</table>
### Feature History for Interfaces in Transparent Mode

#### Gigabit Ethernet Support for the ASA 5510 Security Plus License
- **Platform Releases:** 7.2(3)
- **Feature Information:** The ASA 5510 now supports GE (Gigabit Ethernet) for port 0 and 1 with the Security Plus license. If you upgrade the license from Base to Security Plus, the capacity of the external Ethernet0/0 and Ethernet0/1 ports increases from the original FE (Fast Ethernet) (100 Mbps) to GE (1000 Mbps). The interface names will remain Ethernet 0/0 and Ethernet 0/1. Use the `speed` command to change the speed on the interface and use the `show interface` command to see what speed is currently configured for each interface.

#### Native VLAN support for the ASA 5505
- **Platform Releases:** 7.2(4)/8.0(4)
- **Feature Information:** You can now include the native VLAN in an ASA 5505 trunk port.
  - We introduced the following command: `switchport trunk native vlan`.

#### Jumbo packet support for the ASA 5580
- **Platform Releases:** 8.1(1)
- **Feature Information:** The Cisco ASA 5580 supports jumbo frames. A jumbo frame is an Ethernet packet larger than the standard maximum of 1518 bytes (including Layer 2 header and FCS), up to 9216 bytes. You can enable support for jumbo frames for all interfaces by increasing the amount of memory to process Ethernet frames. Assigning more memory for jumbo frames might limit the maximum use of other features, such as access lists.
  - We introduced the following command: `jumbo-frame reservation`.

#### Increased VLANs for the ASA 5580
- **Platform Releases:** 8.1(2)
- **Feature Information:** The number of VLANs supported on the ASA 5580 are increased from 100 to 250.

#### IPv6 support for transparent mode
- **Platform Releases:** 8.2(1)
- **Feature Information:** IPv6 support was introduced for transparent firewall mode.

#### Support for Pause Frames for Flow Control on the ASA 5580 10-Gigabit Ethernet Interfaces
- **Platform Releases:** 8.2(2)
- **Feature Information:** You can now enable pause (XOFF) frames for flow control.
  - We introduced the following command: `flowcontrol`.

#### Bridge groups for transparent mode
- **Platform Releases:** 8.4(1)
- **Feature Information:** If you do not want the overhead of security contexts, or want to maximize your use of security contexts, you can group interfaces together in a bridge group, and then configure multiple bridge groups, one for each network. Bridge group traffic is isolated from other bridge groups. You can configure up to eight bridge groups of four interfaces each in single mode or per context.
  - We introduced the following commands: `interface bvi`, `show bridge-group`.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gigabit Ethernet Support for the ASA 5510 Security Plus License</td>
<td>7.2(3)</td>
<td>The ASA 5510 now supports GE (Gigabit Ethernet) for port 0 and 1 with the Security Plus license. If you upgrade the license from Base to Security Plus, the capacity of the external Ethernet0/0 and Ethernet0/1 ports increases from the original FE (Fast Ethernet) (100 Mbps) to GE (1000 Mbps). The interface names will remain Ethernet 0/0 and Ethernet 0/1. Use the <code>speed</code> command to change the speed on the interface and use the <code>show interface</code> command to see what speed is currently configured for each interface.</td>
</tr>
<tr>
<td>Native VLAN support for the ASA 5505</td>
<td>7.2(4)/8.0(4)</td>
<td>You can now include the native VLAN in an ASA 5505 trunk port.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>We introduced the following command: <code>switchport trunk native vlan</code>.</td>
</tr>
<tr>
<td>Jumbo packet support for the ASA 5580</td>
<td>8.1(1)</td>
<td>The Cisco ASA 5580 supports jumbo frames. A jumbo frame is an Ethernet packet larger than the standard maximum of 1518 bytes (including Layer 2 header and FCS), up to 9216 bytes. You can enable support for jumbo frames for all interfaces by increasing the amount of memory to process Ethernet frames. Assigning more memory for jumbo frames might limit the maximum use of other features, such as access lists.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>We introduced the following command: <code>jumbo-frame reservation</code>.</td>
</tr>
<tr>
<td>Increased VLANs for the ASA 5580</td>
<td>8.1(2)</td>
<td>The number of VLANs supported on the ASA 5580 are increased from 100 to 250.</td>
</tr>
<tr>
<td>IPv6 support for transparent mode</td>
<td>8.2(1)</td>
<td>IPv6 support was introduced for transparent firewall mode.</td>
</tr>
<tr>
<td>Support for Pause Frames for Flow Control on the ASA 5580 10-Gigabit Ethernet Interfaces</td>
<td>8.2(2)</td>
<td>You can now enable pause (XOFF) frames for flow control.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>We introduced the following command: <code>flowcontrol</code>.</td>
</tr>
<tr>
<td>Bridge groups for transparent mode</td>
<td>8.4(1)</td>
<td>If you do not want the overhead of security contexts, or want to maximize your use of security contexts, you can group interfaces together in a bridge group, and then configure multiple bridge groups, one for each network. Bridge group traffic is isolated from other bridge groups. You can configure up to eight bridge groups of four interfaces each in single mode or per context.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>We introduced the following commands: <code>interface bvi</code>, <code>show bridge-group</code>.</td>
</tr>
</tbody>
</table>
PART 4

Configuring Basic Settings
Configuring Basic Settings

This chapter describes how to configure basic settings on your ASASM that are typically required for a functioning configuration. This chapter includes the following sections:

- Configuring the Hostname, Domain Name, and Passwords, page 9-1
- Configuring the Master Passphrase, page 9-3
- Configuring the DNS Server, page 9-8

Configuring the Hostname, Domain Name, and Passwords

This section describes how to change the device name and passwords, and includes the following topics:

- Changing the Login Password, page 9-1
- Changing the Enable Password, page 9-2
- Setting the Hostname, page 9-2
- Setting the Domain Name, page 9-3

Changing the Login Password

To change the login password, enter the following command:

```
(password | password) password
```

Changes the login password. The login password is used for Telnet and SSH connections. The default login password is “cisco.”

You can enter `passwd` or `password`. The password is a case-sensitive password of up to 16 alphanumeric and special characters. You can use any character in the password except a question mark or a space.

The password is saved in the configuration in encrypted form, so you cannot view the original password after you enter it. Use the `no password` command to restore the password to the default setting.
Changing the Enable Password

To change the enable password, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>enable password password</code></td>
<td>Changes the enable password, which lets you enter privileged EXEC mode. By default, the enable password is blank.</td>
</tr>
</tbody>
</table>

**Example:**

```
hostname(config)# passwd Pa$$w0rd
```

- The `password` argument is a case-sensitive password of up to 16 alphanumeric and special characters. You can use any character in the password except a question mark or a space.
- This command changes the password for the highest privilege level. If you configure local command authorization, you can set enable passwords for each privilege level from 0 to 15.
- The password is saved in the configuration in encrypted form, so you cannot view the original password after you enter it. Enter the `enable password` command without a password to set the password to the default, which is blank.

Setting the Hostname

To set the hostname, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>hostname name</code></td>
<td>Specifies the hostname for the ASASM or for a context.</td>
</tr>
</tbody>
</table>

**Example:**

```
hostname(config)# hostname farscape
farscape(config)#
```

- This name can be up to 63 characters. A hostname must start and end with a letter or digit, and have as interior characters only letters, digits, or a hyphen.
- When you set a hostname for the ASASM, that name appears in the command line prompt. If you establish sessions to multiple devices, the hostname helps you keep track of where you enter commands. The default hostname depends on your platform.
- For multiple context mode, the hostname that you set in the system execution space appears in the command line prompt for all contexts. The hostname that you optionally set within a context does not appear in the command line, but can be used by the `banner` command `$hostname` token.
Setting the Domain Name

To set the domain name, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>domain-name name</td>
<td>Specifies the domain name for the ASASM.</td>
</tr>
<tr>
<td></td>
<td>The ASASM appends the domain name as a suffix to unqualified names.</td>
</tr>
<tr>
<td></td>
<td>For example, if you set the domain name to “example.com,” and specify</td>
</tr>
<tr>
<td></td>
<td>a syslog server by the unqualified name of “jupiter,” then the ASASM</td>
</tr>
<tr>
<td></td>
<td>qualifies the name to “jupiter.example.com.”</td>
</tr>
<tr>
<td></td>
<td>The default domain name is default.domain.invalid.</td>
</tr>
<tr>
<td></td>
<td>For multiple context mode, you can set the domain name for each context,</td>
</tr>
<tr>
<td></td>
<td>as well as within the system execution space.</td>
</tr>
</tbody>
</table>

Configuring the Master Passphrase

This section describes how to configure the master passphrase and includes the following topics:

- Information About the Master Passphrase, page 9-3
- Licensing Requirements for the Master Passphrase, page 9-4
- Guidelines and Limitations, page 9-4
- Adding or Changing the Master Passphrase, page 9-4
- Disabling the Master Passphrase, page 9-6
- Recovering the Master Passphrase, page 9-7
- Feature History for the Master Passphrase, page 9-8

Information About the Master Passphrase

The master passphrase feature allows you to securely store plain text passwords in encrypted format. The master passphrase provides a key that is used to universally encrypt or mask all passwords, without changing any functionality. Features that implement the master passphrase include the following:

- OSPF
- EIGRP
- VPN load balancing
- VPN (remote access and site-to-site)
- Failover
- AAA servers
- Logging
- Shared licenses
Licensing Requirements for the Master Passphrase

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

Context Mode Guidelines

Supported in single and multiple context mode.

Adding or Changing the Master Passphrase

This section describes how to add or change the master passphrase.

Prerequisites

- If failover is enabled but no failover shared key is set, an error message appears if you change the master passphrase, informing you that you must enter a failover shared key to protect the master passphrase changes from being sent as plain text.
- This procedure will only be accepted in a secure session, for example by console, SSH, or ASDM via HTTPS.

To add or change the master passphrase, perform the following steps:
## Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** | **key config-key password-encryption**  
[new_passphrase [old_passphrase]] | Sets the passphrase used for generating the encryption key. The passphrase must be between 8 and 128 characters long. All characters except a back space and double quotes are accepted for the passphrase.  
If you do not enter the new passphrase in the command, you are prompted for it.  
When you want to change the passphrase, you also have to enter the old passphrase.  
See the “Examples” section on page 9-6 for examples of the interactive prompts.  
**Note** Use the interactive prompts to enter passwords to avoid having the passwords logged in the command history buffer.  
Use the **no key config-key password-encrypt** command with caution, because it changes the encrypted passwords into plain text passwords. You can use the no form of this command when downgrading to a software version that does not support password encryption. |
| **Step 2** | **password encryption aes** | Enables password encryption. As soon as password encryption is turned on and the master passphrase is available, all the user passwords will be encrypted. The running configuration will show the passwords in the encrypted format.  
If the passphrase is not configured at the time that password encryption is enabled, the command will succeed in anticipation that the passphrase will be available in the future.  
If you later disable password encryption using the **no password encryption aes** command, all existing encrypted passwords are left unchanged, and as long as the master passphrase exists, the encrypted passwords will be decrypted, as required by the application. |
| **Step 3** | **write memory** | Saves the runtime value of the master passphrase and the resulting configuration. If you do not enter this command, passwords in startup configuration may still be visible if they were not saved with encryption before.  
In addition, in multiple context mode the master passphrase is changed in the system context configuration. As a result, the passwords in all contexts will be affected. If the **write memory** command is not entered in the system context mode, but not in all user contexts, then the encrypted passwords in user contexts may be stale. Alternatively, use the **write memory all** command in the system context to save all configurations. |

**Example:**

```
hostname(config)# key config-key
password-encryption
Old key: bumblebee
New key: haverford
Confirm key: haverford
```

```
hostname(config)# password encryption aes
```

```
hostname(config)# write memory
```
Examples

In the following configuration example, no previous key is present:

hostname (config)# key config-key password-encryption 12345678

In the following configuration example, a key already exists:

Hostname (config)# key config-key password-encryption 23456789
Old key: 12345678
hostname (config)#

In the following configuration example, you want to key in interactively, but a key already exists. The Old key, New key, and Confirm key prompts will appear on your screen if you enter the `key config-key password-encryption` command and press Enter to access interactive mode.

hostname (config)# key config-key password-encryption
Old key: 12345678
New key: 23456789
Confirm key: 23456789

In the following example, you want to key in interactively, but no key is present. The New key and Confirm key prompts will appear on your screen if you are in interactive mode.

hostname (config)# key config-key password-encryption
New key: 12345678
Confirm key: 12345678

Disabling the Master Passphrase

Disabling the master passphrase reverts encrypted passwords into plain text passwords. Removing the passphrase might be useful if you downgrade to a previous software version that does not support encrypted passwords.

Prerequisites

- You must know the current master passphrase to disable it. If you do not know the passphrase, see the “Recovering the Master Passphrase” section on page 9-7.
- This procedure will only be accepted in a secure session, that is, by Telnet, SSH, or ASDM via HTTPS.
Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> no key config-key password-encryption [old_passphrase]</td>
<td>Removes the master passphrase. If you do not enter the passphrase in the command, you are prompted for it.</td>
</tr>
<tr>
<td><strong>Example:</strong> hostname(config)# no key config-key password-encryption</td>
<td><strong>Warning:</strong> You have chosen to revert the encrypted passwords to plain text. This operation will expose passwords in the configuration and therefore exercise caution while viewing, storing, and copying configuration.</td>
</tr>
<tr>
<td></td>
<td>Old key: bumblebee</td>
</tr>
<tr>
<td><strong>Step 2</strong> write memory</td>
<td>Saves the run time value of the master passphrase and the resulting configuration. The non-volatile memory containing the passphrase will be erased and overwritten with the 0xFF pattern.</td>
</tr>
<tr>
<td><strong>Example:</strong> hostname(config)# write memory</td>
<td>In multiple mode the master passphrase is changed in the system context configuration. As a result the passwords in all contexts will be affected. If the write memory command is not entered in the system context mode, but not in all user contexts, then the encrypted passwords in user contexts may be stale. Alternatively, use the write memory all command in the system context to save all configurations.</td>
</tr>
</tbody>
</table>

### Recovering the Master Passphrase

You cannot recover the master passphrase. If the master passphrase is lost or unknown, you can remove it using the write erase command followed by the reload command. These commands remove the master key and the configuration that includes the encrypted passwords.
Feature History for the Master Passphrase

Table 9-1 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Master Passphrase</td>
<td>8.3(1)</td>
<td>This feature was introduced. We introduced the following commands: <code>key config-key password-encryption</code>, <code>password encryption aes</code>, <code>clear configure password encryption aes</code>, <code>show running-config password encryption aes</code>, <code>show password encryption</code>.</td>
</tr>
<tr>
<td>Password Encryption Visibility</td>
<td>8.4(1)</td>
<td>We modified the <code>show password encryption</code> command.</td>
</tr>
</tbody>
</table>

Configuring the DNS Server

Some ASASM features require use of a DNS server to access external servers by domain name; for example, the Botnet Traffic Filter feature requires a DNS server to access the dynamic database server and to resolve entries in the static database. Other features, such as the `ping` or `traceroute` command, let you enter a name that you want to ping or traceroute, and the ASASM can resolve the name by communicating with a DNS server. Many SSL VPN and certificate commands also support names.

Note

The ASASM has limited support for using the DNS server, depending on the feature. For example, most commands require you to enter an IP address and can only use a name when you manually configure the `name` command to associate a name with an IP address and enable use of the names using the `names` command.

For information about dynamic DNS, see the “Configuring DDNS” section on page 11-2.

Prerequisites

Make sure that you configure the appropriate routing for any interface on which you enable DNS domain lookup so you can reach the DNS server. See the “Information About Routing” section on page 19-1 for more information about routing.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>dns domain-lookup</code></td>
<td>Enables the ASASM to send DNS requests to a DNS server to perform a name lookup for supported commands.</td>
</tr>
<tr>
<td><code>interface_name</code></td>
<td></td>
</tr>
</tbody>
</table>

Example:

```
hostname(config)# dns domain-lookup inside
```
Monitoring DNS Cache

The ASASM provides a local cache of DNS information from external DNS queries that are sent for certain clientless SSL VPN and certificate commands. Each DNS translation request is first looked for in the local cache. If the local cache has the information, the resulting IP address is returned. If the local cache can not resolve the request, a DNS query is sent to the various DNS servers that have been configured. If an external DNS server resolves the request, the resulting IP address is stored in the local cache with its corresponding hostname.

DNS Cache Monitoring Commands

To monitor the DNS cache, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show dns-hosts</td>
<td>Show the DNS cache, which includes dynamically learned entries from a DNS server as well as manually entered name and IP addresses using the name command.</td>
</tr>
</tbody>
</table>

Feature History for DNS Cache

Table 2 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>DNS Cache</td>
<td>7.0(1)</td>
<td>DNS cache stores responses that allow a DNS server to respond more quickly to queries. We introduced the following command: show dns host.</td>
</tr>
</tbody>
</table>
Configuring DHCP

This chapter describes how to configure the DHCP server and includes the following sections:

- Information About DHCP, page 10-1
- Licensing Requirements for DHCP, page 10-1
- Guidelines and Limitations, page 10-2
- Configuring a DHCP Server, page 10-2
- Configuring DHCP Relay Services, page 10-7
- DHCP Monitoring Commands, page 10-8
- Feature History for DHCP, page 10-8

Information About DHCP

DHCP provides network configuration parameters, such as IP addresses, to DHCP clients. The ASASM can provide a DHCP server or DHCP relay services to DHCP clients attached to ASASM interfaces. The DHCP server provides network configuration parameters directly to DHCP clients. DHCP relay passes DHCP requests received on one interface to an external DHCP server located behind a different interface.

Licensing Requirements for DHCP

Table 10-1 shows the licensing requirements for DHCP.

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

For the ASA 5505, the maximum number of DHCP client addresses varies depending on the license:

- If the limit is 10 hosts, the maximum available DHCP pool is 32 addresses.
- If the limit is 50 hosts, the maximum available DHCP pool is 128 addresses.
- If the number of hosts is unlimited, the maximum available DHCP pool is 256 addresses.
Guidelines and Limitations

Use the following guidelines to configure the DHCP server:

- You can configure a DHCP server on each interface of the ASASM. Each interface can have its own pool of addresses to draw from. However, the other DHCP settings, such as DNS servers, domain name, options, ping timeout, and WINS servers, are configured globally and used by the DHCP server on all interfaces.

- You cannot configure a DHCP client or DHCP relay services on an interface on which the server is enabled. Additionally, DHCP clients must be directly connected to the interface on which the server is enabled.

- The ASASM does not support QIP DHCP servers for use with DHCP proxy.

- The relay agent cannot be enabled if the DHCP server is also enabled.

- When it receives a DHCP request, the ASASM sends a discovery message to the DHCP server. This message includes the IP address (within a subnetwork) configured with the `dhcp-network-scope` command in the group policy. If the server has an address pool that falls within that subnetwork, the server sends the offer message with the pool information to the IP address—not to the source IP address of the discovery message.

- For example, if the server has a pool in the range of 209.165.200.225 to 209.165.200.254, mask 255.255.255.0, and the IP address specified by the `dhcp-network-scope` command is 209.165.200.1, the server sends that pool in the offer message to the ASASM.

Failover Guidelines
Supports Active/Active and Active/Standby failover.

Firewall Mode Guidelines
Supported in routed and transparent firewall modes.

Context Mode Guidelines
Supported in single mode and multiple context mode.

Configuring a DHCP Server

This section describes how to configure a DHCP server provided by the ASASM and includes the following topics:

- Enabling the DHCP Server, page 10-3
- Configuring DHCP Options, page 10-4
- Using Cisco IP Phones with a DHCP Server, page 10-6
- DHCP Monitoring Commands, page 10-8

By default, the ASA 5505 ships with a 10-user license.
## Enabling the DHCP Server

The ASASM can act as a DHCP server. DHCP is a protocol that provides network settings to hosts, including the host IP address, the default gateway, and a DNS server.

**Note**

The ASASM DHCP server does not support BOOTP requests. In multiple context mode, you cannot enable the DHCP server or DHCP relay on an interface that is used by more than one context.

To enable the DHCP server on a ASASM interface, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>dhcpd address ip_address-ip_address interface_name</td>
<td>Create a DHCP address pool. The ASASM assigns a client one of the addresses from this pool to use for a given length of time. These addresses are the local, untranslated addresses for the directly connected network. The address pool must be on the same subnet as the ASASM interface.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# dhcpd address 10.0.1.101-10.0.1.110 inside</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>dhcpd dns dns1 [dns2]</td>
<td>(Optional) Specifies the IP address(es) of the DNS server(s).</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# dhcpd dns 209.165.201.2 209.165.202.129</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>dhcpd wins wins1 [wins2]</td>
<td>(Optional) Specifies the IP address(es) of the WINS server(s). You can specify up to two WINS servers.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# dhcpd wins 209.165.201.5</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>dhcpd lease lease_length</td>
<td>(Optional) Change the lease length to be granted to the client. This lease equals the amount of time (in seconds) the client can use its allocated IP address before the lease expires. Enter a value between 0 to 1,048,575. The default value is 3600 seconds.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# dhcpd lease 3000</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td>dhcpd domain domain_name</td>
<td>(Optional) Configures the domain name.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# dhcpd domain example.com</td>
<td></td>
</tr>
<tr>
<td><strong>Step 6</strong></td>
<td>dhcpd ping_timeout milliseconds</td>
<td>(Optional) Configures the DHCP ping timeout value. To avoid address conflicts, the ASASM sends two ICMP ping packets to an address before assigning that address to a DHCP client. This command specifies the timeout value for those packets.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# dhcpd ping timeout 20</td>
<td></td>
</tr>
</tbody>
</table>
## Configuring DHCP Options

You can configure the ASASM to send information for the DHCP options listed in RFC 2132. The DHCP options include the following three categories:

- Options that Return an IP Address, page 10-4
- Options that Return a Text String, page 10-4
- Options that Return a Hexadecimal Value, page 10-5

The ASASM supports all three categories. To configure a DHCP option, choose one of the following commands:

### Options that Return an IP Address

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| `dhcpd option code ip addr_1 [addr_2]` | Configures a DHCP option that returns one or two IP addresses.  

**Example:**
```
hostname(config)# dhcpd option 2 ip 10.10.1.1 10.10.1.2
```

### Options that Return a Text String

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| `dhcpd option code ascii text` | Configures a DHCP option that returns a text string.  

**Example:**
```
hostname(config)# dhcpd option 2 ascii examplestring
```
Options that Return a Hexadecimal Value

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>dhcpd option code hex value</td>
<td>Configures a DHCP option that returns a hexadecimal value.</td>
</tr>
</tbody>
</table>

Example:

```plaintext
hostname(config)# dhcpd option 2 hex
22.0011.01.FF1111.00FF.0000.AAAA.1111.1111.1111.11
```

The ASASM does not verify that the option type and value that you provide match the expected type and value for the option code as defined in RFC 2132. For example, you can enter the dhcpd option 46 ascii hello command, and the ASASM accepts the configuration, although option 46 is defined in RFC 2132 to expect a single-digit, hexadecimal value. For more information about the option codes and their associated types and expected values, see RFC 2132.

Table 10-2 shows the DHCP options that are not supported by the `dhcpd option` command.

### Table 10-2 Unsupported DHCP Options

<table>
<thead>
<tr>
<th>Option Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>DHCPOPT_PAD</td>
</tr>
<tr>
<td>1</td>
<td>HCPOPT_SUBNET_MASK</td>
</tr>
<tr>
<td>12</td>
<td>DHCPOPT_HOST_NAME</td>
</tr>
<tr>
<td>50</td>
<td>DHCPOPT_REQUESTED_ADDRESS</td>
</tr>
<tr>
<td>51</td>
<td>DHCPOPT_LEASE_TIME</td>
</tr>
<tr>
<td>52</td>
<td>DHCPOPT_OPTION_OVERLOAD</td>
</tr>
<tr>
<td>53</td>
<td>DHCPOPT_MESSAGE_TYPE</td>
</tr>
<tr>
<td>54</td>
<td>DHCPOPT_SERVER_IDENTIFIER</td>
</tr>
<tr>
<td>58</td>
<td>DHCPOPT_RENEWAL_TIME</td>
</tr>
<tr>
<td>59</td>
<td>DHCPOPT_REBINDING_TIME</td>
</tr>
<tr>
<td>61</td>
<td>DHCPOPT_CLIENT_IDENTIFIER</td>
</tr>
<tr>
<td>67</td>
<td>DHCPOPT_BOOT_FILE_NAME</td>
</tr>
<tr>
<td>82</td>
<td>DHCPOPT_RELAY_INFORMATION</td>
</tr>
<tr>
<td>255</td>
<td>DHCPOPT_END</td>
</tr>
</tbody>
</table>

DHCP options 3, 66, and 150 are used to configure Cisco IP Phones. For more information about configuring these options, see the “Using Cisco IP Phones with a DHCP Server” section on page 10-6.
Using Cisco IP Phones with a DHCP Server

Enterprises with small branch offices that implement a Cisco IP Telephony Voice over IP solution typically implement Cisco CallManager at a central office to control Cisco IP Phones at small branch offices. This implementation allows centralized call processing, reduces the equipment required, and eliminates the administration of additional Cisco CallManager and other servers at branch offices.

Cisco IP Phones download their configuration from a TFTP server. When a Cisco IP Phone starts, if it does not have both the IP address and TFTP server IP address preconfigured, it sends a request with option 150 or 66 to the DHCP server to obtain this information.

- DHCP option 150 provides the IP addresses of a list of TFTP servers.
- DHCP option 66 gives the IP address or the hostname of a single TFTP server.

Note

Cisco IP Phones might also include DHCP option 3 in their requests, which sets the default route.

A single request might include both options 150 and 66. In this case, the ASASM DHCP server provides values for both options in the response if they are already configured on the ASASM.

You can configure the ASASM to send information for most options listed in RFC 2132. The following examples show the syntax for any option number, as well as the syntax for options 3, 66, and 150:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>dhcpd option number value</td>
<td>Provides information for DHCP requests that include an option number as specified in RFC-2132.</td>
</tr>
</tbody>
</table>

**Example:**

```
hostname(config)# dhcpd option 2
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>dhcpd option 66 ascii server_name</td>
<td>Provides the IP address or name of a TFTP server for option 66.</td>
</tr>
</tbody>
</table>

**Example:**

```
hostname(config)# dhcpd option 66 ascii exampleserver
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>dhcpd option 150 ip [server_ip1] [server_ip2]</td>
<td>Provides the IP address or names of one or two TFTP servers for option 150. The server_ip1 is the IP address or name of the primary TFTP server while server_ip2 is the IP address or name of the secondary TFTP server. A maximum of two TFTP servers can be identified using option 150.</td>
</tr>
</tbody>
</table>

**Example:**

```
hostname(config)# dhcpd option 150 ip 10.10.1.1
```
Configuring DHCP Relay Services

A DHCP relay agent allows the ASASM to forward DHCP requests from clients to a router connected to a different interface.

The following restrictions apply to the use of the DHCP relay agent:

- The relay agent cannot be enabled if the DHCP server feature is also enabled.
- DHCP clients must be directly connected to the ASASM and cannot send requests through another relay agent or a router.
- For multiple context mode, you cannot enable DHCP relay on an interface that is used by more than one context.
- DHCP Relay services are not available in transparent firewall mode. An ASASM in transparent firewall mode only allows ARP traffic through; all other traffic requires an access list. To allow DHCP requests and replies through the ASASM in transparent mode, you need to configure two access lists, one that allows DHCP requests from the inside interface to the outside, and one that allows the replies from the server in the other direction.
- When DHCP relay is enabled and more than one DHCP relay server is defined, the ASASM forwards client requests to each defined DHCP relay server. Replies from the servers are also forwarded to the client until the client DHCP relay binding is removed. The binding is removed when the ASASM receives any of the following DHCP messages: ACK, NACK, or decline.

Note

You cannot enable DHCP Relay on an interface running DHCP Proxy. You must Remove VPN DHCP configuration first or you will see an error message. This error happens if both DHCP relay and DHCP proxy are enabled. Ensure that either DHCP relay or DHCP proxy are enabled, but not both.

To enable DHCP relay, perform the following steps:

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>dhcprelay server ip_address if_name</td>
<td>Set the IP address of a DHCP server on a different interface from the DHCP client. You can use this command up to ten times to identify up to ten servers.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# dhcprelay server 201.168.200.4 outside</td>
<td></td>
</tr>
<tr>
<td>Step 2</td>
<td>dhcprelay enable interface</td>
<td>Enables DHCP relay on the interface connected to the clients.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# dhcprelay enable inside</td>
<td></td>
</tr>
</tbody>
</table>
Chapter 10      Configuring DHCP

DHCP Monitoring Commands

To monitor DHCP, enter one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show running-config dhcpd</td>
<td>Shows the current DHCP configuration.</td>
</tr>
<tr>
<td>show running-config dhcprelay</td>
<td>Shows the current DHCP relay services status.</td>
</tr>
</tbody>
</table>

Feature History for DHCP

Table 10-3 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DHCP</td>
<td>7.0(1)</td>
<td>The ASASM can provide a DHCP server or DHCP relay services to DHCP clients attached to ASASM interfaces. We introduced the following commands: dhcp client update dns, dhcpd address, dhcpd domain, dhcpd enable, dhcpd lease, dhcpd option, dhcpd ping timeout, dhcpd update dns, dhcpd wins, dhcp-network-scope, dhcprelay enable, dhcprelay server, dhcprelay setroute, dhcprelay trusted, dhcp-server, show running-config dhcpd, and show running-config dhcprelay.</td>
</tr>
</tbody>
</table>
Configuring Dynamic DNS

This chapter describes how to configure DDNS update methods and includes the following topics:

- Information About DDNS, page 11-1
- Licensing Requirements for DDNS, page 11-2
- Guidelines and Limitations, page 11-2
- Configuring DDNS, page 11-2
- Configuration Examples for DDNS, page 11-3
- DDNS Monitoring Commands, page 11-6
- Feature History for DDNS, page 11-6

Information About DDNS

DDNS update integrates DNS with DHCP. The two protocols are complementary: DHCP centralizes and automates IP address allocation; DDNS update automatically records the association between assigned addresses and hostnames at pre-defined intervals. DDNS allows frequently changing address-hostname associations to be updated frequently. Mobile hosts, for example, can then move freely on a network without user or administrator intervention. DDNS provides the necessary dynamic update and synchronization of the name-to-address mapping and address-to-name mapping on the DNS server. To configure the DNS server for other uses, see the “Configuring the DNS Server” section on page 9-8. To configure DHCP, see the “Configuring a DHCP Server” section on page 10-2.

EDNS allows DNS requesters to advertise the size of their UDP packets and facilitates the transfer of packets larger than 512 octets. When a DNS server receives a request over UDP, it identifies the size of the UDP packet from the OPT resource record (RR) and scales its response to contain as many resource records as are allowed in the maximum UDP packet size specified by the requester. The size of the DNS packets can be up to 4096 bytes for BIND or 1280 bytes for the Windows 2003 DNS Server. Several additional message-length maximum commands are available:

- The existing global limit: message-length maximum 512
- A client or server specific limit: message-length maximum client 4096
- The dynamic value specified in the OPT RR field: message-length maximum client auto

If the three commands are present at the same time, the ASASM enforces the minimum of the three specified values.
Chapter 11  Configuring Dynamic DNS

Licensing Requirements for DDNS

The following table shows the licensing requirements for DDNS:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

Guidelines and Limitations

Failover Guidelines
Supports Active/Active and Active/Standby failover.

Firewall Mode Guidelines
Supported in routed firewall mode.

Context Mode Guidelines
Supported in single and multiple context modes.
Supported in transparent mode for the DNS Client pane.

IPv6 Guidelines
Supports IPv6.

Configuring DDNS

This section describes examples for configuring the ASASM to support Dynamic DNS. DDNS update integrates DNS with DHCP. The two protocols are complementary—DHCP centralizes and automates IP address allocation, while dynamic DNS update automatically records the association between assigned addresses and hostnames. When you use DHCP and dynamic DNS update, this configures a host automatically for network access whenever it attaches to the IP network. You can locate and reach the host using its permanent, unique DNS hostname. Mobile hosts, for example, can move freely without user or administrator intervention.

DDNS provides address and domain name mapping so that hosts can find each other, even though their DHCP-assigned IP addresses change frequently. The DDNS name and address mapping is held on the DHCP server in two resource records: the A RR includes the name-to IP-address mapping, while the PTR RR maps addresses to names. Of the two methods for performing DDNS updates—the IETF standard defined by RFC 2136 and a generic HTTP method—the ASASM supports the IETF method in this release.

The two most common DDNS update configurations are the following:
- The DHCP client updates the A RR, while the DHCP server updates the PTR RR.
- The DHCP server updates both the A RR and PTR RR.
In general, the DHCP server maintains DNS PTR RRs on behalf of clients. Clients may be configured to perform all desired DNS updates. The server may be configured to honor these updates or not. To update the PTR RR, the DHCP server must know the FQDN of the client. The client provides an FQDN to the server using a DHCP option called Client FQDN.

**Configuration Examples for DDNS**

The following examples present five common scenarios:

- **Example 1: Client Updates Both A and PTR RRs for Static IP Addresses**, page 11-3
- **Example 2: Client Updates Both A and PTR RRs; DHCP Server Honors Client Update Request; FQDN Provided Through Configuration**, page 11-3
- **Example 3: Client Includes FQDN Option Instructing Server Not to Update Either RR; Server Overrides Client and Updates Both RRs.**, page 11-4
- **Example 4: Client Asks Server To Perform Both Updates; Server Configured to Update PTR RR Only; Honors Client Request and Updates Both A and PTR RR**, page 11-5
- **Example 5: Client Updates A RR; Server Updates PTR RR**, page 11-5

**Example 1: Client Updates Both A and PTR RRs for Static IP Addresses**

The following example shows how to configure the client to request that it update both A and PTR resource records for static IP addresses.

To configure this scenario, perform the following steps:

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>To define a DDNS update method called ddns-2 that requests that the client update both the A RR and PTR RR, enter the following commands:</td>
</tr>
<tr>
<td></td>
<td>hostname(config)# ddns update method ddns-2</td>
</tr>
<tr>
<td></td>
<td>hostname(DDNS-update-method)# ddns both</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>To associate the method ddns-2 with the eth1 interface, enter the following commands:</td>
</tr>
<tr>
<td></td>
<td>hostname(DDNS-update-method)# interface eth1</td>
</tr>
<tr>
<td></td>
<td>hostname(config-if)# ddns update ddns-2</td>
</tr>
<tr>
<td></td>
<td>hostname(config-if)# ddns update hostname asa.example.com</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>To configure a static IP address for eth1, enter the following command:</td>
</tr>
<tr>
<td></td>
<td>hostname(config-if)# ip address 10.0.0.40 255.255.255.0</td>
</tr>
</tbody>
</table>

**Example 2: Client Updates Both A and PTR RRs; DHCP Server Honors Client Update Request; FQDN Provided Through Configuration**

The following example shows how to configure the DHCP client to request that it update both the A and PTR RRs, and the DHCP server to honor these requests.

To configure this scenario, perform the following steps:
Step 1  To configure the DHCP client to request that the DHCP server perform no updates, enter the following command:

hostname(config)# dhcp-client update dns server none

Step 2  To create a DDNS update method named ddns-2 on the DHCP client that requests that the client perform both A and PTR updates, enter the following commands:

hostname(config)# ddns update method ddns-2
hostname([DDNS-update-method])# ddns both

Step 3  To associate the method named ddns-2 with the ASASM interface named Ethernet0, and enable DHCP on the interface, enter the following commands:

hostname([DDNS-update-method])# interface Ethernet0
hostname([if-config])# ddns update ddns-2
hostname([if-config])# ddns update hostname asa.example.com
hostname([if-config])# ip address dhcp

Step 4  To configure the DHCP server, enter the following command:

hostname([if-config])# dhcpd update dns

---

**Example 3: Client Includes FQDN Option Instructing Server Not to Update Either RR; Server Overrides Client and Updates Both RRs.**

The following example shows how to configure the DHCP client to include the FQDN option that instruct the DHCP server not to honor either the A or PTR updates. The example also shows how to configure the server to override the client request. As a result, the client does not perform any updates. To configure this scenario, perform the following steps:

Step 1  To configure the update method named ddns-2 to request that it make both A and PTR RR updates, enter the following commands:

hostname(config)# ddns update method ddns-2
hostname([DDNS-update-method])# ddns both

Step 2  To assign the DDNS update method named ddns-2 on interface Ethernet0 and provide the client hostname (asa), enter the following commands:

hostname([DDNS-update-method])# interface Ethernet0
hostname([if-config])# ddns update ddns-2
hostname([if-config])# ddns update hostname asa.example.com

Step 3  To enable the DHCP client feature on the interface, enter the following commands:

hostname([if-config])# dhcp client update dns server none
hostname([if-config])# ip address dhcp

Step 4  To configure the DHCP server to override the client update requests, enter the following command:

hostname([if-config])# dhcpd update dns both override
Example 4: Client Asks Server To Perform Both Updates; Server Configured to Update PTR RR Only; Honors Client Request and Updates Both A and PTR RR

The following example shows how to configure the server to perform only PTR RR updates by default. However, the server honors the client request that it perform both A and PTR updates. The server also forms the FQDN by appending the domain name (example.com) to the hostname that the client (asa) has provided.

To configure this scenario, perform the following steps:

**Step 1** To configure the DHCP client on interface Ethernet0, enter the following commands:

```
hostname(config)# interface Ethernet0
hostname(config-if)# dhcp client update dns both
hostname(config-if)# ddns update hostname asa
```

**Step 2** To configure the DHCP server, enter the following commands:

```
hostname(config-if)# dhcpd update dns
hostname(config-if)# dhcpd domain example.com
```

Example 5: Client Updates A RR; Server Updates PTR RR

The following example shows how to configure the client to update the A resource record and how to configure the server to update the PTR records. Also, the client uses the domain name from the DHCP server to form the FQDN.

To configure this scenario, perform the following steps:

**Step 1** To define the DDNS update method named ddns-2, enter the following commands:

```
hostname(config)# ddns update method ddns-2
hostname(DDNS-update-method)# ddns
```

**Step 2** To configure the DHCP client for interface Ethernet0 and assign the update method to the interface, enter the following commands:

```
hostname(DDNS-update-method)# interface Ethernet0
hostname(config-if)# dhcp client update dns
hostname(config-if)# ddns update ddns-2
hostname(config-if)# ddns update hostname asa
```

**Step 3** To configure the DHCP server, enter the following commands:

```
hostname(config-if)# dhcpd update dns
hostname(config-if)# dhcpd domain example.com
```
DDNS Monitoring Commands

To monitor DDNS, enter one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show running-config ddns</code></td>
<td>Shows the current DDNS configuration.</td>
</tr>
<tr>
<td><code>show running-config dns server-group</code></td>
<td>Shows the current DNS server group status.</td>
</tr>
</tbody>
</table>

Feature History for DDNS

Table 11-1 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>DDNS</td>
<td>7.0(1)</td>
<td>This feature was introduced. The following commands were introduced: <strong>ddns</strong>, <strong>ddns update</strong>, <strong>dhcp client update</strong>, <strong>dns</strong>, <strong>dhcpd update dns</strong>, <strong>show running-config ddns</strong>, and <strong>show running-config dns server-group</strong>.</td>
</tr>
</tbody>
</table>
PART 5

Configuring Objects and Access Lists
Objects are reusable components for use in your configuration. They can be defined and used in ASASM configurations in the place of inline IP addresses. Objects make it easy to maintain your configurations because you can modify an object in one place and have it be reflected in all other places that are referencing it. Without objects you would have to modify the parameters for every feature when required, instead of just once. For example, if a network object defines an IP address and subnet mask, and you want to change the address, you only need to change it in the object definition, not in every feature that refers to that IP address.

This chapter describes how to configure objects, and it includes the following sections:

- Configuring Objects and Groups, page 12-1
- Configuring Regular Expressions, page 12-12
- Scheduling Extended Access List Activation, page 12-16

### Configuring Objects and Groups

This section includes the following topics:

- Information About Objects and Groups, page 12-1
- Licensing Requirements for Objects and Groups, page 12-2
- Guidelines and Limitations for Objects and Groups, page 12-3
- Configuring Objects, page 12-3
- Configuring Object Groups, page 12-6
- Monitoring Objects and Groups, page 12-11
- Feature History for Objects and Groups, page 12-12

### Information About Objects and Groups

The ASASM supports objects and object groups. You can attach or detach objects from one or more object groups when needed, ensuring that the objects are not duplicated but can be re-used wherever needed.

This section includes the following topics:
• Information About Objects, page 12-2
• Information About Object Groups, page 12-2

Information About Objects

Objects are created in and used by the ASASM in the place of an inline IP address in any given configuration. You can define an object with a particular IP address and netmask pair or a protocol (and, optionally, a port) and use this object in several configurations. The advantage is that whenever you want to modify the configurations created to this IP address or protocol, you do not need to modify all rules in the running configuration. You can modify the object, and then the change automatically applies to all rules that use the specified object. You can configure two types of objects: network objects and service objects. These objects can be used in Network Address Translation (NAT), access lists, and object groups.

Information About Object Groups

By grouping like objects together, you can use the object group in an ACE instead of having to enter an ACE for each object separately. You can create the following types of object groups:

• Protocol
• Network
• Service
• ICMP type

For example, consider the following three object groups:

• MyServices—Includes the TCP and UDP port numbers of the service requests that are allowed access to the internal network.
• TrustedHosts—Includes the host and network addresses allowed access to the greatest range of services and servers.
• PublicServers—Includes the host addresses of servers to which the greatest access is provided.

After creating these groups, you could use a single ACE to allow trusted hosts to make specific service requests to a group of public servers.

You can also nest object groups in other object groups.

Licensing Requirements for Objects and Groups

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>
Guidelines and Limitations for Objects and Groups

This section includes the guidelines and limitations for this feature.

Context Mode Guidelines
Supported in single and multiple context mode.

Firewall Mode Guidelines
Supported in routed and transparent firewall modes.

IPv6 Guidelines
Supports IPv6, with limitations. (See the “Additional Guidelines and Limitations” section on page 12-3.)

Additional Guidelines and Limitations
The following guidelines and limitations apply to object groups:

- Objects and object groups share the same name space.
- Object groups must have unique names. While you might want to create a network object group named “Engineering” and a service object group named “Engineering,” you need to add an identifier (or “tag”) to the end of at least one object group name to make it unique. For example, you can use the names “Engineering_admins” and “Engineering_hosts” to make the object group names unique and to aid in identification.
- You cannot remove an object group or make an object group empty if it is used in a command.
- The ASASM does not support IPv6 nested object groups, so you cannot group an object with IPv6 entities under another IPv6 object group.

Configuring Objects

This section includes the following topics:

- Configuring a Network Object, page 12-3
- Configuring a Service Object, page 12-4

Configuring a Network Object

A network object contains a single IP address/mask pair. Network objects can be of three types: host, subnet, or range.

You can also configure auto NAT as part of the object definition; see Chapter 28, “Configuring Network Object NAT,” for more information.
Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1**  
object network `obj_name` | Creates a new network object. The `obj_name` is a text string up to 64 characters in length and can be any combination of letters, digits, and the following characters:  
- underscore “_”  
- dash “-”  
- period “.”  
The prompt changes to network object configuration mode. |
| Example:  
hostname(config)# object-network OBJECT1 | |
| **Step 2**  
(host `ip_addr` | subnet `net_addr net_mask` | range `ip_addr_1 ip_addr_2`) | Assigns the IP address to the named object. You can configure a host address, a subnet, or a range of addresses. |
| Example:  
hostname(config-network-object)# host 10.2.2.2 | |
| **Step 3**  
description `text` | Adds a description to the object. |
| Example:  
hostname(config-network-object)#  
description Engineering Network | |

**Examples**

To create a network object, enter the following commands:

```
hostname (config)# object network OBJECT1
hostname (config-network-object)# host 10.2.2.2
```

**Configuring a Service Object**

A service object contains a protocol and optional source and/or destination port.
Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><strong>object service</strong> <em>obj_name</em></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><strong>hostname(config)# object-service SERVOBJECT1</strong></td>
</tr>
<tr>
<td></td>
<td>Creates a new service object. The <em>obj_name</em> is a text string up to 64 characters in length and can be any combination of letters, digits, and the following characters:</td>
</tr>
<tr>
<td></td>
<td>• underscore “_”</td>
</tr>
<tr>
<td></td>
<td>• dash “-”</td>
</tr>
<tr>
<td></td>
<td>• period “.”</td>
</tr>
<tr>
<td></td>
<td>The prompt changes to service object configuration mode.</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><strong>service</strong> *(protocol</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><strong>hostname(config-service-object)# service tcp source eq www destination eq ssh</strong></td>
</tr>
<tr>
<td></td>
<td>Creates a service object for the source mapped address.</td>
</tr>
<tr>
<td></td>
<td>The <em>protocol</em> argument specifies an IP protocol name or number.</td>
</tr>
<tr>
<td></td>
<td>The <em>icmp</em>, <em>tcp</em>, or <em>udp</em> keywords specify that this service object is for either the ICMP, TCP, or UDP protocol.</td>
</tr>
<tr>
<td></td>
<td>The <em>icmp-type</em> argument names the ICMP type.</td>
</tr>
<tr>
<td></td>
<td>The <em>icmp6</em> keyword specifies that the service type is for ICMP version 6 connections.</td>
</tr>
<tr>
<td></td>
<td>The <em>icmp6-type</em> argument names the ICMP version 6 type.</td>
</tr>
<tr>
<td></td>
<td>The <em>source</em> keyword specifies the source port.</td>
</tr>
<tr>
<td></td>
<td>The <em>destination</em> keyword specifies the destination port.</td>
</tr>
<tr>
<td></td>
<td>The <em>operator port</em> argument specifies a single port/code value that supports configuring the port for the protocol. You can specify “eq,” “neq,” “lt,” “gt,” and “range” when configuring a port for TCP or UDP. The “range” operator lists the beginning port and ending port.</td>
</tr>
</tbody>
</table>

**Example**

To create a service object, enter the following commands:

```
hostname (config)# object service SERVOBJECT1
hostname (config-service-object)# service tcp source eq www destination eq ssh
```
**Configuring Object Groups**

This section includes the following topics:

- Adding a Protocol Object Group, page 12-6
- Adding a Network Object Group, page 12-7
- Adding a Service Object Group, page 12-8
- Adding an ICMP Type Object Group, page 12-9
- Nesting Object Groups, page 12-10
- Removing Object Groups, page 12-11

**Adding a Protocol Object Group**

To add or change a protocol object group, perform the steps in this section. After you add the group, you can add more objects as required by following this procedure again for the same group name and specifying additional objects. You do not need to reenter existing objects; the commands you already set remain in place unless you remove them with the `no` form of the command.

### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** | **object-group protocol obj_grp_id** | Adds a protocol group. The `obj_grp_id` is a text string up to 64 characters in length and can be any combination of letters, digits, and the following characters:  
- underscore “_”  
- dash “-”  
- period “.”  
The prompt changes to protocol configuration mode. |
| **Example:** | **hostname(config)# object-group protocol tcp_udp_icmp** |

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 2</strong></td>
<td><strong>description text</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><strong>hostname(config-protocol)# description New Group</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 3</strong></td>
<td><strong>protocol-object protocol</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><strong>hostname(config-protocol)# protocol-object tcp</strong></td>
</tr>
</tbody>
</table>

### Example

To create a protocol group for TCP, UDP, and ICMP, enter the following commands:

```
hostname (config)# object-group protocol tcp_udp_icmp
hostname (config-protocol)# protocol-object tcp
hostname (config-protocol)# protocol-object udp
```
Adding a Network Object Group

A network object group supports IPv4 and IPv6 addresses.

To add or change a network object group, perform the steps in this section. After you add the group, you can add more objects as required by following this procedure again for the same group name and specifying additional objects. You do not need to reenter existing objects; the commands you already set remain in place unless you remove them with the no form of the command.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td><code>object-group network grp_id</code></td>
<td>Adds a network group. The <code>grp_id</code> is a text string up to 64 characters in length and can be any combination of letters, digits, and the following characters: underscore &quot;_&quot;, dash &quot;-&quot;, period &quot;.&quot; The prompt changes to protocol configuration mode.</td>
</tr>
<tr>
<td>Example: <code>hostname(config)# object-group network admins</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td><code>description text</code></td>
<td>(Optional) Adds a description. The description can be up to 200 characters.</td>
</tr>
<tr>
<td>Example: <code>hostname(config-network)# Administrator Addresses</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td></td>
</tr>
<tr>
<td>`network-object {object name</td>
<td>host ip_address</td>
</tr>
<tr>
<td>Example: <code>hostname(config-network)# network-object host 10.2.2.4</code></td>
<td></td>
</tr>
</tbody>
</table>

Example

To create a network group that includes the IP addresses of three administrators, enter the following commands:

```
hostname (config)# object-group network admins
hostname (config-protocol)# description Administrator Addresses
hostname (config-protocol)# network-object host 10.2.2.4
hostname (config-protocol)# network-object host 10.2.2.78
hostname (config-protocol)# network-object host 10.2.2.34
```
Adding a Service Object Group

To add or change a service object group, perform the steps in this section. After you add the group, you can add more objects as required by following this procedure again for the same group name and specifying additional objects. You do not need to reenter existing objects; the commands you already set remain in place unless you remove them with the no form of the command.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>**object-group service grp_id (tcp</td>
</tr>
<tr>
<td><strong>Example</strong></td>
<td>hostname(config)# object-group service services1 tcp-udp</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><strong>description text</strong></td>
</tr>
<tr>
<td><strong>Example</strong></td>
<td>hostname(config-service)# description DNS Group</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>**port-object {eq port</td>
</tr>
<tr>
<td><strong>Example</strong></td>
<td>hostname(config-service)# port-object eq domain</td>
</tr>
</tbody>
</table>

**Command Purpose**

- **Step 1**
  - **object-group service grp_id (tcp | udp | tcp-udp)**
  - Adds a service group.
  - The `object` keyword adds an additional object to the service object group.
  - The `grp_id` is a text string up to 64 characters in length and can be any combination of letters, digits, and the following characters:
    - underscore “_”
    - dash “-”
    - period “.”
  - Specify the protocol for the services (ports) you want to add with either the `tcp`, `udp`, or `tcp-udp` keywords. Enter the `tcp-udp` keyword if your service uses both TCP and UDP with the same port number, for example, DNS (port 53).
  - The prompt changes to service configuration mode.

- **Step 2**
  - **description text**
  - (Optional) Adds a description. The description can be up to 200 characters.

- **Step 3**
  - **port-object {eq port | range begin_port end_port}**
  - Defines the ports in the group. Enter the command for each port or range of ports. For a list of permitted keywords and well-known port assignments, see the “Protocols and Applications” section on page B-11.

**Example**

To create service groups that include DNS (TCP/UDP), LDAP (TCP), and RADIUS (UDP), enter the following commands:

```
hostname (config)# object-group service services1 tcp-udp
hname (config-service)# description DNS Group
hostname (config-service)# port-object eq domain

hostname (config)# object-group service services2 udp
hostname (config-service)# description RADIUS Group
hostname (config-service)# port-object eq radius
```
hostname (config-service)# `port-object eq radius-acct
hostname (config)# `object-group service services3 tcp
hostname (config-service)# `description LDAP Group
hostname (config-service)# `port-object eq ldap

### Adding an ICMP Type Object Group

To add or change an ICMP type object group, perform the steps in this section. After you add the group, you can add more objects as required by following this procedure again for the same group name and specifying additional objects. You do not need to reenter existing objects; the commands you already set remain in place unless you remove them with the `no` form of the command.

#### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** `object-group icmp-type grp_id` | Adds an ICMP type object group. The `grp_id` is a text string up to 64 characters in length and can be any combination of letters, digits, and the following characters:  
  - underscore “_”  
  - dash “-”  
  - period “.”  
  The prompt changes to ICMP type configuration mode. |
| **Example:**  
  hostname(config)# `object-group icmp-type ping` |  |
| **Step 2** `description text` | (Optional) Adds a description. The description can be up to 200 characters. |
| **Example:**  
  hostname(config-icmp-type)# `description Ping Group` |  |
| **Step 3** `icmp-object icmp-type` | Defines the ICMP types in the group. Enter the command for each type. For a list of ICMP types, see the “ICMP Types” section on page B-15. |
| **Example:**  
  hostname(config-icmp-type)# `icmp-object echo-reply` |  |

### Example

Create an ICMP type group that includes echo-reply and echo (for controlling ping) by entering the following commands:

hostname (config)# `object-group icmp-type ping
hostname (config-service)# `description Ping Group
hostname (config-service)# `icmp-object echo
hostname (config-service)# `icmp-object echo-reply
Nesting Object Groups

You can nest object groups hierarchically so that one object group can contain other object groups of the same type and you can mix and match nested group objects and regular objects within an object group. The ASASM does not support IPv6 nested object groups, however, so you cannot group an object with IPv6 entities under another IPv6 object-group.

To nest an object group within another object group of the same type, first create the group that you want to nest (see the “Configuring Object Groups” section on page 12-6), and then perform the steps in this section.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td></td>
</tr>
<tr>
<td>object-group group {{ protocol</td>
<td>network</td>
</tr>
</tbody>
</table>

Example:

```
hostname(config)# object-group network Engineering_group
```

Step 2  | group-object group_id |

Example:

```
hostname(config-network)# group-object Engineering_groups
```

Examples

Create network object groups for privileged users from various departments by entering the following commands:

```
hostname (config)# object-group network eng
hostname (config-network)# network-object host 10.1.1.5
hostname (config-network)# network-object host 10.1.1.9
hostname (config-network)# network-object host 10.1.1.89
hostname (config)# object-group network hr
hostname (config-network)# network-object host 10.1.2.8
hostname (config-network)# network-object host 10.1.2.12
hostname (config)# object-group network finance
hostname (config-network)# network-object host 10.1.4.89
hostname (config-network)# network-object host 10.1.4.100
```

You then nest all three groups together as follows:

```
hostname (config)# object-group network admin
hostname (config-network)# group-object eng
hostname (config-network)# group-object hr
hostname (config-network)# group-object finance
```
You only need to specify the admin object group in your ACE as follows:

```
hostname (config)# access-list ACL_IN extended permit ip object-group admin host
209.165.201.29
```

### Removing Object Groups

You can remove a specific object group or remove all object groups of a specified type; however, you cannot remove an object group or make an object group empty if it is used in an access list.

---

**Detailed Step**

**Step 1**

Do one of the following:

- `no object-group grp_id`

  **Example:**
  ```
  hostname(config)# no object-group Engineering_host
  ```

  Removes the specified object group. The `grp_id` is a text string up to 64 characters in length and can be any combination of letters, digits, and the following characters:
  - underscore “_”
  - dash “-”
  - period “.”

- `clear object-group [protocol | network | services | icmp-type]`

  **Example:**
  ```
  hostname(config)# clear-object group network
  ```

  Removes all object groups of the specified type.

  **Note** If you do not enter a type, all object groups are removed.

---

### Monitoring Objects and Groups

To monitor objects and groups, enter the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show access-list</code></td>
<td>Displays the access list entries that are expanded out into individual entries without their object groupings.</td>
</tr>
<tr>
<td><code>show running-config object-group</code></td>
<td>Displays all current object groups.</td>
</tr>
<tr>
<td><code>show running-config object-group grp_id</code></td>
<td>Displays the current object groups by their group ID.</td>
</tr>
<tr>
<td><code>show running-config object-group grp_type</code></td>
<td>Displays the current object groups by their group type.</td>
</tr>
</tbody>
</table>
Feature History for Objects and Groups

Table 1 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Object groups</td>
<td>7.0(1)</td>
<td>Object groups simplify access list creation and maintenance. We introduced or modified the following commands: <code>object-group protocol</code>, <code>object-group network</code>, <code>object-group service</code>, <code>object-group icmp_type</code>.</td>
</tr>
<tr>
<td>Objects</td>
<td>8.3(1)</td>
<td>Object support was introduced. We introduced or modified the following commands: <code>object-network</code>, <code>object-service</code>, <code>object-group network</code>, <code>object-group service</code>, <code>network object</code>, <code>access-list extended</code>, <code>access-list webtype</code>, <code>access-list remark</code>.</td>
</tr>
</tbody>
</table>

Configuring Regular Expressions

A regular expression matches text strings either literally as an exact string, or by using metacharacters so that you can match multiple variants of a text string. You can use a regular expression to match the content of certain application traffic; for example, you can match a URL string inside an HTTP packet. This section describes how to create a regular expression and includes the following topics:

- Creating a Regular Expression, page 12-12
- Creating a Regular Expression Class Map, page 12-15

Creating a Regular Expression

A regular expression matches text strings either literally as an exact string, or by using metacharacters so you can match multiple variants of a text string. You can use a regular expression to match the content of certain application traffic; for example, you can match a URL string inside an HTTP packet.

Guidelines

Use Ctrl+V to escape all of the special characters in the CLI, such as question mark (?) or a tab. For example, type `d[Ctrl+V]?g` to enter `d?g` in the configuration.

See the `regex` command in the command reference for performance impact information when matching a regular expression to packets.

Note

As an optimization, the ASASM searches on the deobfuscated URL. Deobfuscation compresses multiple forward slashes (/) into a single slash. For strings that commonly use double slashes, like “http://”, be sure to search for “http://” instead.
Table 12-2 lists the metacharacters that have special meanings.

<table>
<thead>
<tr>
<th>Character</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>.</td>
<td>Dot</td>
<td>Matches any single character. For example, <code>d.g</code> matches <code>dog</code>, <code>dag</code>, <code>dtg</code>, and any word that contains those characters, such as <code>doggonit</code>.</td>
</tr>
<tr>
<td>(exp)</td>
<td>Subexpression</td>
<td>A subexpression segregates characters from surrounding characters, so that you can use other metacharacters on the subexpression. For example, `d(o</td>
</tr>
<tr>
<td></td>
<td>Alternation</td>
<td>Matches either expression it separates. For example, <code>dogcat</code> matches dog or cat.</td>
</tr>
</tbody>
</table>
| ?         | Question mark | A quantifier that indicates that there are 0 or 1 of the previous expression. For example, `lo?se` matches lse or lose.  
**Note** You must enter `Ctrl+V` and then the question mark or else the help function is invoked. |
| *         | Asterisk     | A quantifier that indicates that there are 0, 1 or any number of the previous expression. For example, `lo*se` matches lse, lose, loose, and so on. |
| +         | Plus         | A quantifier that indicates that there is at least 1 of the previous expression. For example, `lo+se` matches lose and loose, but not lse. |
| \{x\} or \{x,\} | Minimum repeat quantifier | Repeat at least x times. For example, `ab(xy){2,}z` matches abxyxyz, abxyxyxyz, and so on. |
| [abc]     | Character class | Matches any character in the brackets. For example, `[abc]` matches a, b, or c. |
|[^abc]     | Negated character class | Matches a single character that is not contained within the brackets. For example, `[^abc]` matches any character other than a, b, or c. `[^A-Z]` matches any single character that is not an uppercase letter. |
| [a-c]     | Character range class | Matches any character in the range. `[a-z]` matches any lowercase letter. You can mix characters and ranges: `[abcq-z]` matches a, b, c, q, r, s, t, u, v, w, x, y, z, and so does `[a-cq-z]`.  
The dash (-) character is literal only if it is the last or the first character within the brackets: `[abc-]` or `[-abc]`. |
| ""       | Quotation marks | Preserves trailing or leading spaces in the string. For example, “test” preserves the leading space when it looks for a match. |
| ^         | Caret        | Specifies the beginning of a line. |
### Detailed Steps

**Step 1** To test a regular expression to make sure it matches what you think it will match, enter the following command:

```
hostname(config)# test regex input_text regular_expression
```

Where the `input_text` argument is a string you want to match using the regular expression, up to 201 characters in length.

The `regular_expression` argument can be up to 100 characters in length.

Use `Ctrl+V` to escape all of the special characters in the CLI. For example, to enter a tab in the input text in the `test regex` command, you must enter `test regex "test[Ctrl+V Tab]" "test\t"`.

If the regular expression matches the input text, you see the following message:

```
INFO: Regular expression match succeeded.
```

If the regular expression does not match the input text, you see the following message:

```
INFO: Regular expression match failed.
```

**Step 2** To add a regular expression after you tested it, enter the following command:

```
hostname(config)# regex name regular_expression
```

Where the `name` argument can be up to 40 characters in length.

The `regular_expression` argument can be up to 100 characters in length.

### Examples

The following example creates two regular expressions for use in an inspection policy map:

```
hostname(config)# regex url_example example\.com
```
Creating a Regular Expression Class Map

A regular expression class map identifies one or more regular expressions. You can use a regular expression class map to match the content of certain traffic; for example, you can match URL strings inside HTTP packets.

Detailed Steps

**Step 1** Create one or more regular expressions according to the “Configuring Regular Expressions” section.

**Step 2** Create a class map by entering the following command:

```
hostname(config)# class-map type regex match-any class_map_name
```

Where `class_map_name` is a string up to 40 characters in length. The name “class-default” is reserved. All types of class maps use the same name space, so you cannot reuse a name already used by another type of class map.

The `match-any` keyword specifies that the traffic matches the class map if it matches at least one of the regular expressions.

The CLI enters class-map configuration mode.

**Step 3** (Optional) Add a description to the class map by entering the following command:

```
hostname(config-cmap)# description string
```

**Step 4** Identify the regular expressions you want to include by entering the following command for each regular expression:

```
hostname(config-cmap)# match regex regex_name
```

Examples

The following example creates two regular expressions, and adds them to a regular expression class map. Traffic matches the class map if it includes the string “example.com” or “example2.com.”

```
hostname(config)# regex url_example example\.com
hostname(config)# regex url_example2 example2\.com
hostname(config)# class-map type regex match-any URLs
hostname(config-cmap)# match regex url_example
hostname(config-cmap)# match regex url_example2
```
Scheduling Extended Access List Activation

This section includes the following topics:

- Information About Scheduling Access List Activation, page 12-16
- Licensing Requirements for Scheduling Access List Activation, page 12-16
- Guidelines and Limitations for Scheduling Access List Activation, page 12-16
- Configuring and Applying Time Ranges, page 12-17
- Configuration Examples for Scheduling Access List Activation, page 12-18
- Feature History for Scheduling Access List Activation, page 12-18

Information About Scheduling Access List Activation

You can schedule each ACE in an access list to be activated at specific times of the day and week by applying a time range to the ACE.

Licensing Requirements for Scheduling Access List Activation

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

Guidelines and Limitations for Scheduling Access List Activation

This section includes the guidelines and limitations for this feature.

Context Mode Guidelines
Supported in single and multiple context mode.

Firewall Mode Guidelines
Supported in routed and transparent firewall modes.

IPv6 Guidelines
Supports IPv6.
### Additional Guidelines and Limitations

The following guidelines and limitations apply to using object groups with access lists:

- Users could experience a delay of approximately 80 to 100 seconds after the specified end time for the ACL to become inactive. For example, if the specified end time is 3:50, because the end time is inclusive, the command is picked up anywhere between 3:51:00 and 3:51:59. After the command is picked up, the ASASM finishes any currently running task and then services the command to deactivate the ACL.

- Multiple periodic entries are allowed per `time-range` command. If a `time-range` command has both `absolute` and `periodic` values specified, then the periodic commands are evaluated only after the absolute start time is reached, and they are not further evaluated after the absolute end time is reached.

### Configuring and Applying Time Ranges

You can add a time range to implement a time-based access list. To identify the time range, perform the steps in this section.

#### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td><code>time-range name</code></td>
<td>Identifies the time-range name.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# time range Sales</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 2</th>
<th>Do one of the following:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><code>periodic days-of-the-week time to</code> [days-of-the-week] time</td>
<td>Specifies a recurring time range.</td>
</tr>
</tbody>
</table>
|        | **Example:** hostname(config-time-range)# periodic | You can specify the following values for `days-of-the-week`:
|        | monday 7:59 to friday 17:01                      |   • `monday`, `tuesday`, `wednesday`, `thursday`, `friday`, `saturday`, or `sunday`. |
|        |                                                    |   • `daily`                                    |
|        |                                                    |   • `weekdays`                                 |
|        |                                                    |   • `weekend`                                  |
|        |                                                    | The `time` is in the format `hh:mm`. For example, 8:00 is 8:00 a.m. and 20:00 is 8:00 p.m. |


### Scheduling Extended Access List Activation

#### Command

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>absolute <strong>start</strong> time date [<strong>end</strong> time date]</td>
<td>Specifies an absolute time range. The <em>time</em> is in the format <em>hh:mm</em>. For example, 8:00 is 8:00 a.m. and 20:00 is 8:00 p.m. The <em>date</em> is in the format <em>day month year</em>; for example, 1 <em>January</em> 2006.</td>
</tr>
<tr>
<td><strong>Step 3</strong> access-list access_list_name [extended] {deny</td>
<td>permit}...{time-range name}</td>
</tr>
</tbody>
</table>

#### Example

The following example binds an access list named “Sales” to a time range named “New_York_Minute”:

```
hostname(config)# access-list Sales line 1 extended deny tcp host 209.165.200.225 host 209.165.201.1 time-range New_York_Minute
```

#### Configuration Examples for Scheduling Access List Activation

The following is an example of an absolute time range beginning at 8:00 a.m. on January 1, 2006. Because no end time and date are specified, the time range is in effect indefinitely.

```
hostname(config)# time-range for2006
hostname(config-time-range)# absolute start 8:00 1 january 2006
```

The following is an example of a weekly periodic time range from 8:00 a.m. to 6:00 p.m on weekdays:

```
hostname(config)# time-range workinghours
hostname(config-time-range)# periodic weekdays 8:00 to 18:00
```

#### Feature History for Scheduling Access List Activation

**Table 12-3** lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scheduling access list activation</td>
<td>7.0</td>
<td>You can schedule each ACE in an access list to be activated at specific times of the day and week. We introduced or modified the following commands: <strong>object-group protocol, object-group network, object-group service, object-group icmp_type.</strong></td>
</tr>
</tbody>
</table>
Information About Access Lists

Cisco ASASMs provide basic traffic filtering capabilities with access lists, which control access in your network by preventing certain traffic from entering or exiting. This chapter describes access lists and shows how to add them to your network configuration.

Access lists are made up of one or more access control entries (ACEs). An ACE is a single entry in an access list that specifies a permit or deny rule (to forward or drop the packet) and is applied to a protocol, to a source and destination IP address or network, and, optionally, to the source and destination ports.

Access lists can be configured for all routed and network protocols (IP, AppleTalk, and so on) to filter the packets of those protocols as the packets pass through a router.

Access lists are used in a variety of features. If your feature uses Modular Policy Framework, you can use an access list to identify traffic within a traffic class map. For more information on Modular Policy Framework, see Chapter 30, “Configuring a Service Policy Using the Modular Policy Framework.”

This chapter includes the following sections:

- Access List Types, page 13-1
- Access Control Entry Order, page 13-2
- Access Control Implicit Deny, page 13-3
- IP Addresses Used for Access Lists When You Use NAT, page 13-3
- Where to Go Next, page 13-3

Access List Types

The ASASM uses five types of access control lists:

- Standard access lists—Identify the destination IP addresses of OSPF routes and can be used in a route map for OSPF redistribution. Standard access lists cannot be applied to interfaces to control traffic. For more information, see Chapter 16, “Adding a Standard Access List.”

- Extended access lists—Use one or more access control entries (ACE) in which you can specify the line number to insert the ACE, the source and destination addresses, and, depending upon the ACE type, the protocol, the ports (for TCP or UDP), or the ICMP type (for ICMP). For more information, see Chapter 14, “Adding an Extended Access List.”

- EtherType access lists—Use one or more ACEs that specify an EtherType. For more information, see Chapter 15, “Adding an EtherType Access List.”

- IPv6 access lists—Determine which IPv6 traffic to block and which traffic to forward at router interfaces. For more information, see Chapter 17, “Adding an IPv6 Access List.”
Table 13-1 lists the types of access lists and some common uses for them.

<table>
<thead>
<tr>
<th>Access List Use</th>
<th>Access List Type</th>
<th>Description</th>
</tr>
</thead>
</table>
| Control network access for IP traffic (routed and transparent mode) | Extended         | The ASASM does not allow any traffic from a lower security interface to a higher security interface unless it is explicitly permitted by an extended access list.  
**Note** To access the ASASM interface for management access, you do not also need an access list allowing the host IP address. You only need to configure management access according to Chapter 34, “Configuring Management Access.” |
| Identify traffic for AAA rules                       | Extended         | AAA rules use access lists to identify traffic.                                                                                                                                                       |
| Control network access for IP traffic for a given user | Extended, downloaded from a AAA server per user | You can configure the RADIUS server to download a dynamic access list to be applied to the user, or the server can send the name of an access list that you already configured on the ASASM. |
| Identify addresses for NAT (policy NAT and NAT exemption) | Extended         | Policy NAT lets you identify local traffic for address translation by specifying the source and destination addresses in an extended access list.                                                   |
| Establish VPN access                                 | Extended         | You can use an extended access list in VPN commands.                                                                                                                                                 |
| Identify traffic in a traffic class map for Modular Policy Framework | Extended, EtherType | Access lists can be used to identify traffic in a class map, which is used for features that support Modular Policy Framework. Features that support Modular Policy Framework include TCP and general connection settings, and inspection. |
| For transparent firewall mode, control network access for non-IP traffic | EtherType         | You can configure an access list that controls traffic based on its EtherType.                                                                                                                         |
| Identify OSPF route redistribution                    | Standard         | Standard access lists include only the destination address. You can use a standard access list to control the redistribution of OSPF routes.                                                          |
| Filtering for WebVPN                                 | Webtype          | You can configure a Webtype access list to filter URLs.                                                                                                                                               |
| Control network access for IPV6 networks             | IPv6             | You can add and apply access lists to control traffic in IPv6 networks.                                                                                                                                |

**Access Control Entry Order**

An access list is made up of one or more access control entries (ACEs). Each ACE that you enter for a given access list name is appended to the end of the access list. Depending on the access list type, you can specify the source and destination addresses, the protocol, the ports (for TCP or UDP), the ICMP type (for ICMP), or the EtherType.

The order of ACEs is important. When the ASASM decides whether to forward or to drop a packet, the ASASM tests the packet against each ACE in the order in which the entries are listed. After a match is found, no more ACEs are checked. For example, if you create an ACE at the beginning of an access list that explicitly permits all traffic, no further statements are checked, and the packet is forwarded.
Access Control Implicit Deny

All access lists have an implicit deny statement at the end, so unless you explicitly permit traffic to pass, it will be denied. For example, if you want to allow all users to access a network through the ASASM except for one or more particular addresses, then you need to deny those particular addresses and then permit all others.

For EtherType access lists, the implicit deny at the end of the access list does not affect IP traffic or ARPs; for example, if you allow EtherType 8037, the implicit deny at the end of the access list does not now block any IP traffic that you previously allowed with an extended access list (or implicitly allowed from a high security interface to a low security interface). However, if you explicitly deny all traffic with an EtherType ACE, then IP and ARP traffic is denied.

IP Addresses Used for Access Lists When You Use NAT

For the following features, you should always use the real IP address in the access list when you use NAT, even if the address as seen on an interface is the mapped address:

- `access-group` command
- Modular Policy Framework `match access-list` command
- Botnet Traffic Filter `dynamic-filter enable classify-list` command
- AAA `aaa ... match` commands
- WCCP `wccp redirect-list group-list` command

The following features use access lists, but these access lists use the mapped values as seen on an interface:

- IPsec access lists
- capture command access lists
- Per-user access lists
- Routing protocols
- All other features...

Where to Go Next

For information about implementing access lists, see the following chapters in this guide:

- Chapter 14, “Adding an Extended Access List”
- Chapter 15, “Adding an EtherType Access List”
- Chapter 16, “Adding a Standard Access List”
- Chapter 17, “Adding an IPv6 Access List”
- Chapter 32, “Configuring Access Rules”
Adding an Extended Access List

This chapter describes how to configure extended access lists (also known as access control lists), and it includes the following sections:

- Information About Extended Access Lists, page 14-1
- Licensing Requirements for Extended Access Lists, page 14-1
- Guidelines and Limitations, page 14-1
- Default Settings, page 14-2
- Configuring Extended Access Lists, page 14-2
- Monitoring Extended Access Lists, page 14-5
- Configuration Examples for Extended Access Lists, page 14-5
- Where to Go Next, page 14-7
- Feature History for Extended Access Lists, page 14-7

Information About Extended Access Lists

Access lists are used to control network access or to specify traffic for many features to act upon. An extended access list is made up of one or more access control entries (ACE) in which you can specify the line number to insert the ACE, the source and destination addresses, and, depending upon the ACE type, the protocol, the ports (for TCP or UDP), or the ICMP type. You can identify all of these parameters within the `access-list` command, or you can use objects for each parameter.

Licensing Requirements for Extended Access Lists

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

This section includes the guidelines and limitations for this feature.

Context Mode Guidelines

Supported in single and multiple context mode.
Firewall Mode Guidelines
Supported only in routed and transparent firewall modes.

IPv6 Guidelines
IPv6 is supported.

Additional Guidelines and Limitations
The following guidelines and limitations apply to creating an extended access list:

- Enter the access list name in uppercase letters so that the name is easy to see in the configuration. You might want to name the access list for the interface (for example, INSIDE), or you can name it for the purpose for which it is created (for example, NO_NAT or VPN).
- Typically, you identify the `ip` keyword for the protocol, but other protocols are accepted. For a list of protocol names, see the “Protocols and Applications” section on page B-11.
- You can specify the source and destination ports only for the TCP or UDP protocols. For a list of permitted keywords and well-known port assignments, see the “TCP and UDP Ports” section on page B-11. DNS, Discard, Echo, Identi, NTP, RPC, SUNRPC, and Talk each require one definition for TCP and one for UDP. TACACS+ requires one definition for port 49 on TCP.
- When you specify a network mask, the method is different from the Cisco IOS software `access-list` command. The ASASM uses a network mask (for example, 255.255.255.0 for a Class C mask). The Cisco IOS mask uses wildcard bits (for example, 0.0.0.255).

Default Settings

Table 14-1 lists the default settings for extended access list parameters.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACE logging</td>
<td>ACE logging generates system log message 106023 for denied packets. A deny ACE must be present to log denied packets.</td>
</tr>
<tr>
<td><code>log</code></td>
<td>When the <code>log</code> keyword is specified, the default level for system log message 106100 is 6 (informational), and the default interval is 300 seconds.</td>
</tr>
</tbody>
</table>

Configuring Extended Access Lists

This section shows how to add and delete an access control entry and access list, and it includes the following topics:

- Adding an Extended Access List, page 14-3
- Adding Remarks to Access Lists, page 14-5
Adding an Extended Access List

An access list is made up of one or more access control entries (ACEs) with the same access list ID. To create an access list you start by creating an ACE and applying a list name. An access list with one entry is still considered a list, although you can add multiple entries to the list.

Prerequisites

(Optional) Create an object or object group according to the “Configuring Objects and Groups” section on page 12-1.

Guidelines

To delete an ACE, enter the `no access-list` command with the entire command syntax string as it appears in the configuration. To remove the entire access list, use the `clear configure access-list` command.
Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>(For IP traffic, no ports)</strong></td>
<td>Adds an extended ACE.</td>
</tr>
<tr>
<td>`access-list access_list_name [line line_number] extended (deny</td>
<td>Adds an extended ACE.</td>
</tr>
<tr>
<td>(protocol</td>
<td>permit) (source_address mask</td>
</tr>
<tr>
<td>object-group</td>
<td><strong>protocol</strong> [object nw_obj_id</td>
</tr>
<tr>
<td><code>object-group</code> nw_grp_id)`</td>
<td>`object-group</td>
</tr>
<tr>
<td><code>object-group</code> nw_grp_id)`</td>
<td>`object-group</td>
</tr>
<tr>
<td>`dest_address mask</td>
<td><strong>protocol</strong> [object nw_obj_id</td>
</tr>
<tr>
<td>object-group</td>
<td><code>object-group</code> nw_grp_id)`</td>
</tr>
<tr>
<td><code>object-group</code> nw_grp_id)`</td>
<td>`object-group</td>
</tr>
<tr>
<td><code>object-group</code> nw_grp_id)`</td>
<td>`object-group</td>
</tr>
<tr>
<td><code>object-group</code> nw_grp_id)`</td>
<td>`object-group</td>
</tr>
<tr>
<td></td>
<td><code>object-group</code> nw_grp_id)`</td>
</tr>
<tr>
<td><code>object-group</code> nw_grp_id)`</td>
<td><code>object-group</code> nw_grp_id)`</td>
</tr>
<tr>
<td><code>object-group</code> nw_grp_id)`</td>
<td><code>object-group</code> nw_grp_id)`</td>
</tr>
<tr>
<td><code>object-group</code> nw_grp_id)`</td>
<td><code>object-group</code> nw_grp_id)`</td>
</tr>
<tr>
<td><code>object-group</code> nw_grp_id)`</td>
<td><code>object-group</code> nw_grp_id)`</td>
</tr>
<tr>
<td><code>object-group</code> nw_grp_id)`</td>
<td><code>object-group</code> nw_grp_id)`</td>
</tr>
</tbody>
</table>

Example:

```
hostname(config)# access-list ACL_IN extended permit ip any any
```
Adding Remarks to Access Lists

You can include remarks about entries in any access list, including extended, EtherType, IPv6, standard, and Webtype access lists. The remarks make the access list easier to understand.

To add a remark after the last `access-list` command you entered, enter the following command:

```
Example
hostname(config)# access-list OUT remark - this is the inside admin address
hostname(config)# access-list OUT extended permit ip host 209.168.200.3 any
hostname(config)# access-list OUT remark - this is the hr admin address
hostname(config)# access-list OUT extended permit ip host 209.168.200.4 any
```

### Command Purpose

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>access-list access_list_name remark text</code></td>
<td>Adds a remark after the last <code>access-list</code> command you entered. The text can be up to 100 characters in length. You can enter leading spaces at the beginning of the text. Trailing spaces are ignored. If you enter the remark before any <code>access-list</code> command, then the remark is the first line in the access list. If you delete an access list using the <code>no access-list access_list_name</code> command, then all the remarks are also removed.</td>
</tr>
</tbody>
</table>

#### Example

You can add remarks before each ACE, and the remark appears in the access list in this location. Entering a dash (-) at the beginning of the remark helps set it apart from the ACEs.

```
hostname(config)# access-list OUT remark - this is the inside admin address
hostname(config)# access-list OUT extended permit ip host 209.168.200.3 any
hostname(config)# access-list OUT remark - this is the hr admin address
hostname(config)# access-list OUT extended permit ip host 209.168.200.4 any
```

### Monitoring Extended Access Lists

To monitor extended access lists, enter one of the following commands:

```
Command | Purpose |
---------|---------|
show access list | Displays the access list entries by number. |
show running-config access-list | Displays the current running access-list configuration. |
```

### Configuration Examples for Extended Access Lists

This section includes the following topics:

- Configuration Examples for Extended Access Lists (No Objects), page 14-6
- Configuration Examples for Extended Access Lists (Using Objects), page 14-6
Configuration Examples for Extended Access Lists (No Objects)

The following access list allows all hosts (on the interface to which you apply the access list) to go through the ASASMs:

```
hostname(config)# access-list ACL_IN extended permit ip any any
```

The following sample access list prevents hosts on 192.168.1.0/24 from accessing the 209.165.201.0/27 network. All other addresses are permitted.

```
hostname(config)# access-list ACL_IN extended deny tcp 192.168.1.0 255.255.255.0 209.165.201.0 255.255.255.224
hostname(config)# access-list ACL_IN extended permit ip any any
```

If you want to restrict access to selected hosts only, then enter a limited permit ACE. By default, all other traffic is denied unless explicitly permitted.

```
hostname(config)# access-list ACL_IN extended permit ip 192.168.1.0 255.255.255.0 209.165.201.0 255.255.255.224
```

The following access list restricts all hosts (on the interface to which you apply the access list) from accessing a website at address 209.165.201.29. All other traffic is allowed.

```
hostname(config)# access-list ACL_IN extended deny tcp any host 209.165.201.29 eq www
hostname(config)# access-list ACL_IN extended permit ip any any
```

The following access list that uses object groups restricts several hosts on the inside network from accessing several web servers. All other traffic is allowed.

```
hostname(config-network)# access-list ACL_IN extended deny tcp object-group denied object-group web eq www
hostname(config)# access-list ACL_IN extended permit ip any any
hostname(config)# access-group ACL_IN in interface inside
```

The following example temporarily disables an access list that permits traffic from one group of network objects (A) to another group of network objects (B):

```
hostname(config)# access-list 104 permit ip host object-group A object-group B inactive
```

To implement a time-based access list, use the `time-range` command to define specific times of the day and week. Then use the `access-list extended` command to bind the time range to an access list. The following example binds an access list named “Sales” to a time range named “New_York_Minute.”

```
hostname(config)# access-list Sales line 1 extended deny tcp host 209.165.200.225 host 209.165.201.1 time-range New_York_Minute
```

Configuration Examples for Extended Access Lists (Using Objects)

The following normal access list that does not use object groups restricts several hosts on the inside network from accessing several web servers. All other traffic is allowed.

```
hostname(config)# access-list ACL_IN extended deny tcp host 10.1.1.4 host 209.165.201.29 eq www
hostname(config)# access-list ACL_IN extended deny tcp host 10.1.1.78 host 209.165.201.29 eq www
hostname(config)# access-list ACL_IN extended deny tcp host 10.1.1.89 host 209.165.201.29 eq www
hostname(config)# access-list ACL_IN extended deny tcp host 10.1.1.4 host 209.165.201.16 eq www
```
hostname(config)# access-list ACL_IN extended deny tcp host 10.1.1.78 host 209.165.201.16 eq www
hostname(config)# access-list ACL_IN extended deny tcp host 10.1.1.89 host 209.165.201.16 eq www
hostname(config)# access-list ACL_IN extended deny tcp host 10.1.1.4 host 209.165.201.78 eq www
hostname(config)# access-list ACL_IN extended deny tcp host 10.1.1.78 host 209.165.201.78 eq www
hostname(config)# access-list ACL_IN extended deny tcp host 10.1.1.89 host 209.165.201.78 eq www
hostname(config)# access-list ACL_IN extended permit ip any any
hostname(config)# access-group ACL_IN in interface inside

If you make two network object groups, one for the inside hosts, and one for the web servers, then the configuration can be simplified and can be easily modified to add more hosts:

hostname(config)# object-group network denied
hostname(config)# object-group network host 10.1.1.4
hostname(config)# object-group network host 10.1.1.78
hostname(config)# object-group network host 10.1.1.89

hostname(config)# object-group network web
hostname(config)# object-group network host 209.165.201.29
hostname(config)# object-group network host 209.165.201.16
hostname(config)# object-group network host 209.165.201.78

hostname(config)# access-list ACL_IN extended deny tcp port object-group denied eq www
hostname(config)# access-list ACL_IN extended permit ip any any
hostname(config)# access-group ACL_IN in interface inside

### Where to Go Next

Apply the access list to an interface. See the “Configuring Access Rules” section on page 32-7 for more information.

### Feature History for Extended Access Lists

Table 14-2 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extended access lists</td>
<td>7.0(1)</td>
<td>Access lists are used to control network access or to specify traffic for many features to act upon. An extended access control list is made up of one or more access control entries (ACE) in which you can specify the line number to insert the ACE, the source and destination addresses, and, depending upon the ACE type, the protocol, the ports (for TCP or UDP), or the ICMP type (for ICMP). We introduced the following command: access-list extended.</td>
</tr>
</tbody>
</table>
Adding an EtherType Access List

This chapter describes how to configure EtherType access lists and includes the following sections:

- Information About EtherType Access Lists, page 15-1
- Licensing Requirements for EtherType Access Lists, page 15-1
- Guidelines and Limitations, page 15-2
- Default Settings, page 15-2
- Configuring EtherType Access Lists, page 15-2
- Monitoring EtherType Access Lists, page 15-4
- What to Do Next, page 15-4
- Configuration Examples for EtherType Access Lists, page 15-5
- Feature History for EtherType Access Lists, page 15-5

Information About EtherType Access Lists

An EtherType access list is made up of one or more Access Control Entries (ACEs) that specify an EtherType. An EtherType rule controls any EtherType identified by a 16-bit hexadecimal number, as well as other traffic types. See the “Supported EtherTypes and Other Traffic” section on page 32-6 for more information.

For information about creating an access rule with the EtherType access list, see Chapter 32, “Configuring Access Rules.”

Licensing Requirements for EtherType Access Lists

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>
Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

**Context Mode Guidelines**
Available in single and multiple context modes.

**Firewall Mode Guidelines**
Supported in transparent firewall mode only.

**IPv6 Guidelines**
Supports IPv6.

**Additional Guidelines and Limitations**
The following guidelines and limitations apply to EtherType access lists:

- For EtherType access lists, the implicit deny at the end of the access list does not affect IP traffic or ARPs; for example, if you allow EtherType 8037, the implicit deny at the end of the access list does not now block any IP traffic that you previously allowed with an extended access list (or implicitly allowed from a high security interface to a low security interface). However, if you *explicitly* deny all traffic with an EtherType ACE, then IP and ARP traffic is denied.

- 802.3-formatted frames are not handled by the access list because they use a length field as opposed to a type field.

- See the “Supported EtherTypes and Other Traffic” section on page 32-6 for more information about supported traffic.

Default Settings

Access list logging generates system log message 106023 for denied packets. Deny packets must be present to log denied packets.

When you configure logging for the access list, the default severity level for system log message 106100 is 6 (informational).

Configuring EtherType Access Lists

This section includes the following topics:

- Task Flow for Configuring EtherType Access Lists, page 15-2
- Adding EtherType Access Lists, page 15-3
- Adding Remarks to Access Lists, page 15-4

Task Flow for Configuring EtherType Access Lists

Use the following guidelines to create and implement an access list:
Step 1  Create an access list by adding an ACE and applying an access list name, as shown in the “Adding EtherType Access Lists” section on page 15-3.

Step 2  Apply the access list to an interface. (See the “Configuring Access Rules” section on page 32-7 for more information.)

Adding EtherType Access Lists

To configure an access list that controls traffic based upon its EtherType, perform the following steps:

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| access-list access_list_name ethertype (deny | permits an EtherType ACE. The *access_list_name* argument lists the name or number of an access list. When you specify an access list name, the ACE is added to the end of the access list. Enter the *access_list_name* in upper case letters so that the name is easy to see in the configuration. You might want to name the access list for the interface (for example, INSIDE) or for the purpose (for example, MPLS or PIX).  

The *deny* keyword denies access if the conditions are matched. If an EtherType access list is configured to deny all, all ethernet frames are discarded. Only physical protocol traffic, such as auto-negotiation, is still allowed.  

The *permit* keyword permits access if the conditions are matched.

The *bpdu* keyword specifies access to bridge protocol data units, which are allowed by default.

The *ipx* keyword specifies access to IPX.

The *mpls-unicast* keyword specifies access to MPLS unicast.

The *mpls-multicast* keyword specifies access to MPLS multicast.

The *any* keyword specifies access for any traffic.

The *hex_number* argument indicates any EtherType that can be identified by a 16-bit hexadecimal number greater than or equal to 0x600. (See RFC 1700, “Assigned Numbers,” at http://www.ietf.org/rfc/rfc1700.txt for a list of EtherTypes.)  

Note  To remove an EtherType ACE, enter the no access-list command with the entire command syntax string as it appears in the configuration.

Example

The following sample access list allows common EtherTypes originating on the inside interface:
Adding Remarks to Access Lists

You can include remarks about entries in any access list, including extended, EtherType, IPv6, standard, and Webtype access lists. The remarks make an access list easier to understand.

To add a remark after the last `access-list` command you entered, enter the following command:

```
access-list access_list_name remark text
```

Example:
```
hostname(config)# access-list OUT remark - this is the inside admin address
hostname(config)# access-list OUT extended permit ip host 209.168.200.3 any
hostname(config)# access-list OUT remark - this is the hr admin address
hostname(config)# access-list OUT extended permit ip host 209.168.200.4 any
```

What to Do Next

Apply the access list to an interface. (See the “Configuring Access Rules” section on page 32-7 for more information.)

Monitoring EtherType Access Lists

To monitor EtherType access lists, enter one of the following commands:

```
Command                  Purpose
show access-list         Displays the access list entries by number.
show running-config access-list Displays the current running access-list configuration.
```
Configuration Examples for EtherType Access Lists

The following example shows how to configure EtherType access lists:

The following access list allows some EtherTypes through the ASASM, but it denies IPX:

```
hostname(config)# access-list ETHER ethertype deny ipx
hostname(config)# access-list ETHER ethertype permit 0x1234
hostname(config)# access-list ETHER ethertype permit mpls-unicast
hostname(config)# access-group ETHER in interface inside
hostname(config)# access-group ETHER in interface outside
```

The following access list denies traffic with EtherType 0x1256, but it allows all others on both interfaces:

```
hostname(config)# access-list nonIP ethertype deny 1256
hostname(config)# access-list nonIP ethertype permit any
hostname(config)# access-group ETHER in interface inside
hostname(config)# access-group ETHER in interface outside
```

Feature History for EtherType Access Lists

Table 15-1 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
</table>
| EtherType access lists | 7.0(1)   | EtherType access lists control traffic based upon its EtherType.  
|                     |          | We introduced the feature and the following command: access-list ethertype. |
Adding a Standard Access List

This chapter describes how to configure a standard access list and includes the following sections:

- Information About Standard Access Lists, page 16-1
- Licensing Requirements for Standard Access Lists, page 16-1
- Guidelines and Limitations, page 16-1
- Default Settings, page 16-2
- Adding Standard Access Lists, page 16-3
- What to Do Next, page 16-4
- Monitoring Access Lists, page 16-4
- Configuration Examples for Standard Access Lists, page 16-4
- Feature History for Standard Access Lists, page 16-5

Information About Standard Access Lists

Standard access lists identify the destination IP addresses of OSPF routes and can be used in a route map for OSPF redistribution. Standard access lists cannot be applied to interfaces to control traffic.

Licensing Requirements for Standard Access Lists

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

Guidelines and Limitations

This section includes the guidelines and limitations for this feature:

- Context Mode Guidelines, page 16-2
- Firewall Mode Guidelines, page 16-2
Default Settings

- Additional Guidelines and Limitations, page 16-2

Context Mode Guidelines
Supported in single context mode only.

Firewall Mode Guidelines
Supported in routed and transparent firewall modes.

IPv6 Guidelines
Supports IPv6.

Additional Guidelines and Limitations
The following guidelines and limitations apply for standard Access Lists:

- Standard ACLs identify the destination IP addresses (not source addresses) of OSPF routes and can be used in a route map for OSPF redistribution. Standard ACLs cannot be applied to interfaces to control traffic.
- To add additional ACEs at the end of the access list, enter another `access-list` command, specifying the same access list name.
- When used with the `access-group` command, the `deny` keyword does not allow a packet to traverse the ASASM. By default, the ASASM denies all packets on the originating interface unless you specifically permit access.
- When specifying a source, local, or destination address, use the following guidelines:
  - Use a 32-bit quantity in four-part, dotted-decimal format.
  - Use the keyword `any` as an abbreviation for an address and mask of 0.0.0.0.0.0.0.0.
  - Use the `host ip_address` option as an abbreviation for a mask of 255.255.255.255.
- You can disable an ACE by specifying the keyword `inactive` in the `access-list` command.

Default Settings

Table 16-1 lists the default settings for standard Access List parameters.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>deny</td>
<td>The ASASM denies all packets on the originating interface unless you specifically permit access. Access list logging generates system log message 106023 for denied packets. Deny packets must be present to log denied packets.</td>
</tr>
</tbody>
</table>
Adding Standard Access Lists

This section includes the following topics:

- Task Flow for Configuring Extended Access Lists, page 16-3
- Adding a Standard Access List, page 16-3
- Adding Remarks to Access Lists, page 16-4

Task Flow for Configuring Extended Access Lists

Use the following guidelines to create and implement an access list:

- Create an access list by adding an ACE and applying an access list name. See in the “Adding Standard Access Lists” section on page 16-3.
- Apply the access list to an interface. See the “Configuring Access Rules” section on page 32-7 for more information.

Adding a Standard Access List

To add an access list to identify the destination IP addresses of OSPF routes, which can be used in a route map for OSPF redistribution, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>hostname(config)# access-list access_list_name standard (deny</td>
<td>permit) {any</td>
</tr>
</tbody>
</table>

Example:

hostname(config)# access-list OSPF standard permit 192.168.1.0 255.255.255.0
Adding Remarks to Access Lists

You can include remarks about entries in any access list, including extended, EtherType, IPv6, standard, and Webtype access lists. The remarks make the access list easier to understand.

To add a remark after the last `access-list` command you entered, enter the following command:

```
access-list access_list_name remark text
```

Example:

```
hostname(config)# access-list OUT remark - this is the inside admin address
```

- **Command Purpose**
  - Adds a remark after the last `access-list` command you entered.
  - The text can be up to 100 characters in length. You can enter leading spaces at the beginning of the text. Trailing spaces are ignored.
  - If you enter the remark before any `access-list` command, then the remark is the first line in the access list.
  - If you delete an access list using the `no access-list access_list_name` command, then all the remarks are also removed.

**Example**

You can add a remark before each ACE, and the remarks appear in the access lists in these locations. Entering a dash (-) at the beginning of a remark helps to set it apart from an ACE.

```
hostname(config)# access-list OUT remark - this is the inside admin address
hostname(config)# access-list OUT extended permit ip host 209.168.200.3 any
hostname(config)# access-list OUT remark - this is the hr admin address
hostname(config)# access-list OUT extended permit ip host 209.168.200.4 any
```

**What to Do Next**

Apply the access list to an interface. See the “Configuring Access Rules” section on page 32-7 for more information.

**Monitoring Access Lists**

To monitor access lists, perform one of the following tasks:

```
Command | Purpose
---------|---------
show access-list | Displays the access list entries by number.
show running-config access-list | Displays the current running access-list configuration.
```

**Configuration Examples for Standard Access Lists**

The following example shows how to deny IP traffic through the ASASM:

```
hostname(config)# access-list 77 standard deny
```
The following example shows how to permit IP traffic through the ASASM if conditions are matched:

```
hostname(config)# access-list 77 standard permit
```

The following example shows how to specify a destination address:

```
hostname(config)# access-list 77 standard permit host 10.1.10.123
```

### Feature History for Standard Access Lists

Table 16-2 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard access lists</td>
<td>7.0(1)</td>
<td>Standard access lists identify the destination IP addresses of OSPF routes, which can be used in a route map for OSPF redistribution. We introduced the feature and the following command: access-list standard.</td>
</tr>
</tbody>
</table>
Adding an IPv6 Access List

This chapter describes how to configure IPv6 access lists to control and filter traffic through the ASASM.

This chapter includes the following sections:
- Information About IPv6 Access Lists, page 17-1
- Licensing Requirements for IPv6 Access Lists, page 17-1
- Prerequisites for Adding IPv6 Access Lists, page 17-2
- Guidelines and Limitations, page 17-2
- Default Settings, page 17-3
- Configuring IPv6 Access Lists, page 17-4
- Monitoring IPv6 Access Lists, page 17-7
- Configuration Examples for IPv6 Access Lists, page 17-7
- Where to Go Next, page 17-7
- Feature History for IPv6 Access Lists, page 17-7

Information About IPv6 Access Lists

The typical access list functionality in IPv6 is similar to access lists in IPv4. Access lists determine which traffic to block and which traffic to forward at router interfaces. Access lists allow filtering based upon source and destination addresses, inbound and outbound to specific interfaces. Each access list has an implicit deny statement at the end. You define IPv6 access lists and set their deny and permit conditions using the `ipv6 access-list` command with the `deny` and `permit` keywords in global configuration mode.

Licensing Requirements for IPv6 Access Lists

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>
Prerequisites for Adding IPv6 Access Lists

You should be familiar with IPv6 addressing and basic configuration. See the `ipv6` commands in the *Cisco Security Appliance Command Reference* for more information about configuring IPv6.

Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

**Context Mode Guidelines**
Supported in single and multiple context modes.

**Firewall Mode Guidelines**
Supported in routed and transparent firewall modes.

**IPv6 Guidelines**
Supports IPv6.

**Additional Guidelines and Limitations**
The following guidelines and limitations apply to IPv6 access lists:

- The `ipv6 access-list` command allows you to specify whether an IPv6 address is permitted or denied access to a port or protocol. Each command is called an ACE. One or more ACEs with the same access list name are referred to as an access list. Apply an access list to an interface using the `access-group` command.
- The ASASM denies all packets from an outside interface to an inside interface unless you specifically permit access using an access list. All packets are allowed by default from an inside interface to an outside interface unless you specifically deny access.
- The `ipv6 access-list` command is similar to the `access-list` command, except that it is IPv6-specific. For additional information about access lists, refer to the `access-list extended` command.
- The `ipv6 access-list icmp` command is used to filter ICMPv6 messages that pass through the ASASM. To configure the ICMPv6 traffic that is allowed to originate and terminate at a specific interface, use the `ipv6 icmp` command.
- See the `object-group` command for information on how to configure object groups.
- Possible operands for the operator option of the `ipv6 access-list` command include `lt` for less than, `gt` for greater than, `eq` for equal to, `neq` for not equal to, and `range` for an inclusive range. Use the `ipv6 access-list` command without an operator and port to indicate all ports by default.
- ICMP message types are filtered by the access rule. Omitting the `icmp_type` argument indicates all ICMP types. If you specify ICMP types, the value can be a valid ICMP type number (from 0 to 255) or one of the following ICMP type literals:
  - destination-unreachable
  - packet-too-big
  - time-exceeded
  - parameter-problem
  - echo-request
- echo-reply
- membership-query
- membership-report
- membership-reduction
- router-renumbering
- router-solicitation
- router-advertisement
- neighbor-solicitation
- neighbor-advertisement
- neighbor-redirect

- If the protocol argument is specified, valid values are `icmp, ip, tcp, udp`, or an integer in the range of 1 to 254, representing an IP protocol number.

**Default Settings**

Table 17-1 lists the default settings for IPv6 access list parameters.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>default</td>
<td>The <code>default</code> option specifies that a syslog message 106100 is generated for the ACE.</td>
</tr>
<tr>
<td>interval <code>secs</code></td>
<td>Specifies the time interval at which to generate a 106100 syslog message; valid values are from 1 to 600 seconds. The default interval is 300 seconds. This value is also used as the timeout value for deleting an inactive flow.</td>
</tr>
<tr>
<td>level</td>
<td>The <code>level</code> option specifies the syslog level for message 106100; valid values are from 0 to 7. The default level is 6 (informational).</td>
</tr>
<tr>
<td>log</td>
<td>The <code>log</code> option specifies logging action for the ACE. If you do not specify the <code>log</code> keyword or you specify the <code>log default</code> keyword, then message 106023 is generated when a packet is denied by the ACE. If you specify the <code>log</code> keyword alone or with a level or interval, then message 106100 is generated when a packet is denied by the ACE. Packets that are denied by the implicit deny at the end of an access list are not logged. You must implicitly deny packets with an ACE to enable logging.</td>
</tr>
</tbody>
</table>
Task Flow for Configuring IPv6 Access Lists

Use the following guidelines to create and implement an access list:

- Create an access list by adding an ACE and applying an access list name, as shown in the “Adding IPv6 Access Lists” section on page 17-5.
- Apply the access list to an interface. (See the “Configuring Access Rules” section on page 32-7 for more information.)
# Adding IPv6 Access Lists

You can add a regular IPv6 access list or add an IPv6 access list with TCP. To add a regular IPv6 access list, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`ipv6 access-list id [line line-num] {deny</td>
<td>permit} {protocol</td>
</tr>
</tbody>
</table>

Example:

```
hostname(config)# ipv6 access-list acl_grp
permit tcp any host 3001:1::203:A0FF:FED6:162D
```
To configure an IPv6 access list with ICMP, enter the following command:

```
ipv6 access-list id \[line line-num\] \{deny \| permit\} icmp\6
\{source-ipv6-prefix/\*prefix-length\ | any \| host source-ipv6-address \| object-group network_obj_grp_id\}
\{destination-ipv6-prefix/\*prefix-length\ | any \| host destination-ipv6-address \| object-group network_obj_grp_id\}
\{icmp_type | object-group icmp_type_obj_grp_id\} \[\log \[\{level\] \{interval secs\ | disable \| default\]\]
```

**Purpose**

Configures an IPv6 access list with ICMP.

The `icmp6` keyword specifies that the access rule applies to ICMPv6 traffic passing through the ASASM.

The `icmp_type` argument specifies the ICMP message type being filtered by the access rule. The value can be a valid ICMP type number from 0 to 255. (For a list of the permitted ICMP type literals, see the “Guidelines and Limitations” section on page 17-2.)

The `icmp_type_obj_grp_id` option specifies the object group ICMP type ID.

For details about additional `ipv6 access-list` command parameters, see the preceding procedure for adding a regular IPv6 access list, or see the `ipv6 access-list` command in the Cisco Security Appliance Command Reference.

---

### Adding Remarks to Access Lists

You can include remarks about entries in any access list, including extended, EtherType, IPv6, standard, and Webtype access lists. The remarks make the access list easier to understand.

To add a remark after the last `access-list` command you entered, enter the following command:

```
access-list access_list_name remark text
```

**Example:**

```
hostname(config)# access-list OUT remark - this is the inside admin address
hostname(config)# access-list OUT remark - this is the hr admin address
```

**Purpose**

Adds a remark after the last `access-list` command you entered.

The text can be up to 100 characters in length. You can enter leading spaces at the beginning of the text. Trailing spaces are ignored.

If you enter the remark before any `access-list` command, then the remark is the first line in the access list.

If you delete an access list using the `no access-list access_list_name` command, then all the remarks are also removed.

---

**Example**

You can add remarks before each ACE, and the remarks appear in the access list in these locations. Entering a dash (-) at the beginning of a remark helps set it apart from an ACE.

```
hostname(config)# access-list OUT remark - this is the inside admin address
hostname(config)# access-list OUT extended permit ip host 209.168.200.3 any
hostname(config)# access-list OUT remark - this is the hr admin address
hostname(config)# access-list OUT extended permit ip host 209.168.200.4 any
```
Monitoring IPv6 Access Lists

To monitor IPv6 access lists, perform one of the following tasks:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show ipv6 access-list</td>
<td>Displays all IPv6 access list information.</td>
</tr>
</tbody>
</table>

Configuration Examples for IPv6 Access Lists

The following example shows how to configure IPv6 access lists:

The following example allows any host using TCP to access the 3001:1::203:A0FF:FED6:162D server:

```
hostname(config)# ipv6 access-list acl_grp permit tcp any host 3001:1::203:A0FF:FED6:162D
```

The following example uses eq and a port to deny access to just FTP:

```
hostname(config)# ipv6 access-list acl_out deny tcp any host 3001:1::203:A0FF:FED6:162D eq ftp
hostname(config)# access-group acl_out in interface inside
```

The following example uses lt to permit access to all ports less than port 2025, which permits access to the well-known ports (1 to 1024):

```
hostname(config)# ipv6 access-list acl_dmz1 permit tcp any host 3001:1::203:A0FF:FED6:162D lt 1025
hostname(config)# access-group acl_dmz1 in interface dmz1
```

Where to Go Next

Apply the access list to an interface. (See the “Configuring Access Rules” section on page 32-7 for more information.)

Feature History for IPv6 Access Lists

Table 17-2 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPv6 access lists</td>
<td>7.0(1)</td>
<td>We introduced the following command: ipv6 access-list.</td>
</tr>
</tbody>
</table>
CHAPTER 18

Configuring Logging for Access Lists

This chapter describes how to configure access list logging for extended access lists and Webtype access lists, and it describes how to manage deny flows.

This chapter includes the following sections:

- Configuring Logging for Access Lists, page 18-1
- Managing Deny Flows, page 18-5

Configuring Logging for Access Lists

This section includes the following topics:

- Information About Logging Access List Activity, page 18-1
- Licensing Requirements for Access List Logging, page 18-2
- Guidelines and Limitations, page 18-2
- Default Settings, page 18-3
- Configuring Access List Logging, page 18-3
- Monitoring Access Lists, page 18-4
- Configuration Examples for Access List Logging, page 18-4
- Feature History for Access List Logging, page 18-5

Information About Logging Access List Activity

By default, when traffic is denied by an extended ACE or a Webtype ACE, the ASASM generates syslog message 106023 for each denied packet in the following form:

```
%ASA|PIX-4-106023: Deny protocol src [interface_name:source_address/source_port] dst interface_name:dest_address/dest_port [type {string}, code {code}] by access_group acl_id
```

If the ASASM is attacked, the number of syslog messages for denied packets can be very large. We recommend that you instead enable logging using syslog message 106100, which provides statistics for each ACE and enables you to limit the number of syslog messages produced. Alternatively, you can disable all logging.
Only ACEs in the access list generate logging messages; the implicit deny at the end of the access list does not generate a message. If you want all denied traffic to generate messages, add the implicit ACE manually to the end of the access list, as shown in the following example:

```
hostname(config)# access-list TEST deny ip any any log
```

The `log` options at the end of the extended `access-list` command enable you to set the following behavior:

- Enable message 106100 instead of message 106023
- Disable all logging
- Return to the default logging using message 106023

Syslog message 106100 uses the following form:

```
%ASA|PIX-n-106100: access-list acl_id {permitted | denied} protocol interface_name/source_address(source_port) -> interface_name/dest_address(dest_port) hit-cnt number {{first hit | number-second interval}}
```

When you enable logging for message 106100, if a packet matches an ACE, the ASASM creates a flow entry to track the number of packets received within a specific interval. The ASASM generates a syslog message at the first hit and at the end of each interval, identifying the total number of hits during the interval and the timestamp for the last hit. At the end of each interval, the ASASM resets the hit count to 0. If no packets match the ACE during an interval, the ASASM deletes the flow entry.

A flow is defined by the source and destination IP addresses, protocols, and ports. Because the source port might differ for a new connection between the same two hosts, you might not see the same flow increment because a new flow was created for the connection. See the “Managing Deny Flows” section on page 18-5 to limit the number of logging flows.

Permitted packets that belong to established connections do not need to be checked against access lists; only the initial packet is logged and included in the hit count. For connectionless protocols, such as ICMP, all packets are logged, even if they are permitted, and all denied packets are logged.

See the `syslog messages guide` guide for detailed information about this syslog message.

### Licensing Requirements for Access List Logging

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

### Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

**Context Mode Guidelines**

Supported in single and multiple context mode.
Firewall Mode Guidelines
Supported only in routed and transparent firewall modes.

IPv6 Guidelines
Supports IPv6.

Additional Guidelines and Limitations
ACE logging generates syslog message 106023 for denied packets. A deny ACE must be present to log denied packets.

Default Settings

Table 18-1 lists the default settings for extended access list parameters.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>log</td>
<td>When the log keyword is specified, the default level for syslog message 106100 is 6 (informational), and the default interval is 300 seconds.</td>
</tr>
</tbody>
</table>

Configuring Access List Logging

This sections describes how to configure access list logging.

Note
For complete access list command syntax, see the “Configuring Extended Access Lists” section on page 14-2.
To configure logging for an ACE, enter the following command:

```
access-list access_list_name [extended] 
{deny | permit}...[log [(level] [interval 
secs] | disable | default)]
```

Example:
```
hostname(config)# access-list outside-acl
permit ip host 10.0.0.0 any log 7 interval 600
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| access-list access_list_name [extended] 
{deny | permit}...[log [(level] [interval 
secs] | disable | default)] | Configures logging for an ACE. |

The `access-list access_list_name` syntax specifies the access list for which you want to configure logging.
The `extended` option adds an ACE.
The `deny` keyword denies a packet if the conditions are matched. Some features do not allow deny ACEs, such as NAT. (See the command documentation for each feature that uses an access list for more information.)
The `permit` keyword permits a packet if the conditions are matched.
If you enter the `log` option without any arguments, you enable syslog message 106100 at the default level (6) and for the default interval (300 seconds). See the following options:
- `level`—A severity level between 0 and 7. The default is 6.
- `interval secs`—The time interval in seconds between syslog messages, from 1 to 600. The default is 300. This value is also used as the timeout value for deleting an inactive flow.
- `disable`—Disables all access list logging.
- `default`—Enables logging to message 106023. This setting is the same as having no `log` option.

(See the `access-list` command in the Cisco Security Appliance Command Reference for more information about command options.)

## Monitoring Access Lists

To monitor access lists, enter one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show access list</td>
<td>Displays the access list entries by number.</td>
</tr>
<tr>
<td>show running-config access-list</td>
<td>Displays the current running access list configuration.</td>
</tr>
</tbody>
</table>

## Configuration Examples for Access List Logging

This section includes sample configurations for logging access lists.

You might configure the following access list:

```
hostname(config)# access-list outside-acl permit ip host 10.10.0.0 any log 7 interval 600
hostname(config)# access-list outside-acl permit ip host 10.255.255.255 any
hostname(config)# access-list outside-acl deny ip any any log 2
hostname(config)# access-group outside-acl in interface outside
```
When the first ACE of outside-acl permits a packet, the ASASM generates the following syslog message:

```
%ASA|PIX-7-106100: access-list outside-acl permitted tcp outside/10.0.0.0(12345) -> inside/192.168.1.1(1357) hit-cnt 1 (first hit)
```

Although 20 additional packets for this connection arrive on the outside interface, the traffic does not have to be checked against the access list, and the hit count does not increase.

If one or more connections by the same host are initiated within the specified 10-minute interval (and the source and destination ports remain the same), then the hit count is incremented by 1, and the following syslog message displays at the end of the 10-minute interval:

```
%ASA|PIX-7-106100: access-list outside-acl permitted tcp outside/10.0.0.0(12345)-> inside/192.168.1.1(1357) hit-cnt 2 (600-second interval)
```

When the third ACE denies a packet, the ASASM generates the following syslog message:

```
%ASA|PIX-2-106100: access-list outside-acl denied ip outside/10.255.255.255(12345) -> inside/192.168.1.1(1357) hit-cnt 1 (first hit)
```

If 20 additional attempts occur within a 5-minute interval (the default), the following syslog message appears at the end of 5 minutes:

```
%ASA|PIX-2-106100: access-list outside-acl denied ip outside/10.255.255.255(12345) -> inside/192.168.1.1(1357) hit-cnt 21 (300-second interval)
```

---

**Managing Deny Flows**

This section includes the following topics:

- Information About Managing Deny Flows, page 18-6
- Licensing Requirements for Managing Deny Flows, page 18-6
- Guidelines and Limitations, page 18-6
- Managing Deny Flows, page 18-7
- Monitoring Deny Flows, page 18-7
- Feature History for Managing Deny Flows, page 18-8

---

**Feature History for Access List Logging**

Table 18-2 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access list logging</td>
<td>7.0(1)</td>
<td>You can enable logging using syslog message 106100, which provides statistics for each ACE and lets you limit the number of syslog messages produced. We introduced the following command: access-list.</td>
</tr>
<tr>
<td>ACL Timestamp</td>
<td>8.3(1)</td>
<td>The ASASM reports the timestamp for the last access rule hit.</td>
</tr>
</tbody>
</table>
### Information About Managing Deny Flows

When you enable logging for message 106100, if a packet matches an ACE, the ASASM creates a flow entry to track the number of packets received within a specific interval. The ASASM has a maximum of 32 K logging flows for ACEs. A large number of flows can exist concurrently at any point of time. To prevent unlimited consumption of memory and CPU resources, the ASASM places a limit on the number of concurrent `deny` flows; the limit is placed on deny flows only (not on permit flows) because they can indicate an attack. When the limit is reached, the ASASM does not create a new deny flow for logging until the existing flows expire.

For example, if someone initiates a DoS attack, the ASASM can create a large number of deny flows in a short period of time. Restricting the number of deny flows prevents unlimited consumption of memory and CPU resources.

When you reach the maximum number of deny flows, the ASASM issues syslog message 106100:

```
%ASA|PIX-1-106101: The number of ACL log deny-flows has reached limit (number).
```

The `access-list alert-interval` command sets the time interval for generating syslog message 106001. Syslog message 106001 alerts you that the ASASM has reached a deny flow maximum. When the deny flow maximum is reached, another syslog message 106001 is generated if at least six seconds have passed since the last 106001 message was generated.

### Licensing Requirements for Managing Deny Flows

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

### Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

**Context Mode Guidelines**
Supported in single and multiple context mode.

**Firewall Mode Guidelines**
Supported only in routed and transparent firewall modes.

**IPv6 Guidelines**
Supports IPv6.

**Additional Guidelines and Limitations**
The ASASM places a limit on the number of concurrent `deny` flows only—not permit flows.
## Default Settings

Table 18-1 lists the default settings for managing deny flows.

### Table 18-3 Default Parameters for Managing Deny Flows

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>numbers</td>
<td>The <em>numbers</em> argument specifies the maximum number of deny flows. The default is 4096.</td>
</tr>
<tr>
<td>secs</td>
<td>The <em>secs</em> argument specifies the time, in seconds, between syslog messages. The default is 300.</td>
</tr>
</tbody>
</table>

## Managing Deny Flows

To configure the maximum number of deny flows and to set the interval between deny flow alert messages (106100), enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>access-list deny-flow-max number</code></td>
<td>Sets the maximum number of deny flows. The <em>numbers</em> argument specifies the maximum number, which can be between 1 and 4096. The default is 4096.</td>
</tr>
</tbody>
</table>

**Example:**

```
hostname(config)# access-list deny-flow-max 3000
```

To set the amount of time between syslog messages (number 106101), which identifies that the maximum number of deny flows was reached, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>access-list alert-interval secs</code></td>
<td>Sets the time, in seconds, between syslog messages. The <em>secs</em> argument specifies the time interval between each deny flow maximum message. Valid values are from 1 to 3600 seconds. The default is 300 seconds.</td>
</tr>
</tbody>
</table>

**Example:**

```
hostname(config)# access-list alert-interval 200
```

## Monitoring Deny Flows

To monitor access lists, enter one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show access-list</code></td>
<td>Displays access list entries by number.</td>
</tr>
<tr>
<td><code>show running-config access-list</code></td>
<td>Displays the current running access list configuration.</td>
</tr>
</tbody>
</table>
Feature History for Managing Deny Flows

Table 18-2 lists each feature change and the platform release in which it was implemented.

Table 18-4 Feature History for Managing Deny Flows

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Managing Deny Flows</td>
<td>7.0(1)</td>
<td>You can configure the maximum number of deny flows and set the interval between deny flow alert messages. We introduced the following commands: access-list deny-flow and access-list alert-interval.</td>
</tr>
</tbody>
</table>
PART 6

Configuring IP Routing
Routing Overview

This chapter describes underlying concepts of how routing behaves within the ASASM, and the routing protocols that are supported.

This chapter includes the following sections:

- Information About Routing, page 19-1
- How Routing Behaves Within the ASA, page 19-4
- Supported Internet Protocols for Routing, page 19-5
- Information About the Routing Table, page 19-6
- Information About IPv6 Support, page 19-9
- Disabling Proxy ARPs, page 19-11

Information About Routing

Routing is the act of moving information across an internetwork from a source to a destination. Along the way, at least one intermediate node typically is encountered. Routing involves two basic activities: determining optimal routing paths and transporting information groups (typically called packets) through an internetwork. In the context of the routing process, the latter of these is referred to as packet switching. Although packet switching is relatively straightforward, path determination can be very complex.

This section includes the following topics:

- Switching, page 19-2
- Path Determination, page 19-2
- Supported Route Types, page 19-2
Switching

Switching algorithms is relatively simple; it is the same for most routing protocols. In most cases, a host determines that it must send a packet to another host. Having acquired a router address by some means, the source host sends a packet addressed specifically to a router physical (Media Access Control [MAC]-layer) address, this time with the protocol (network layer) address of the destination host.

As it examines the packet destination protocol address, the router determines that it either knows or does not know how to forward the packet to the next hop. If the router does not know how to forward the packet, it typically drops the packet. If the router knows how to forward the packet, however, it changes the destination physical address to that of the next hop and transmits the packet.

The next hop may be the ultimate destination host. If not, the next hop is usually another router, which executes the same switching decision process. As the packet moves through the internetwork, its physical address changes, but its protocol address remains constant.

Path Determination

Routing protocols use metrics to evaluate what path will be the best for a packet to travel. A metric is a standard of measurement, such as path bandwidth, that is used by routing algorithms to determine the optimal path to a destination. To aid the process of path determination, routing algorithms initialize and maintain routing tables, which include route information. Route information varies depending on the routing algorithm used.

Routing algorithms fill routing tables with a variety of information. Destination or next hop associations tell a router that a particular destination can be reached optimally by sending the packet to a particular router representing the next hop on the way to the final destination. When a router receives an incoming packet, it checks the destination address and attempts to associate this address with a next hop.

Routing tables also can include other information, such as data about the desirability of a path. Routers compare metrics to determine optimal routes, and these metrics differ depending on the design of the routing algorithm used.

Routers communicate with one another and maintain their routing tables through the transmission of a variety of messages. The routing update message is one such message that generally consists of all or a portion of a routing table. By analyzing routing updates from all other routers, a router can build a detailed picture of network topology. A link-state advertisement, another example of a message sent between routers, informs other routers of the state of the sender links. Link information also can be used to build a complete picture of network topology to enable routers to determine optimal routes to network destinations.

Asymmetric routing is only supported for Active/Active failover in multiple context mode. For more information, see the “Configuring Active/Active Failover” section on page 51-8.

Supported Route Types

There are several route types that a router can use. The ASASM uses the following route types:

- Static Versus Dynamic, page 19-3
- Single-Path Versus Multipath, page 19-3
- Flat Versus Hierarchical, page 19-3
Static Versus Dynamic

Static routing algorithms are hardly algorithms at all, but are table mappings established by the network administrator before the beginning of routing. These mappings do not change unless the network administrator alters them. Algorithms that use static routes are simple to design and work well in environments where network traffic is relatively predictable and where network design is relatively simple.

Because static routing systems cannot react to network changes, they generally are considered unsuitable for large, constantly changing networks. Most of the dominant routing algorithms are dynamic routing algorithms, which adjust to changing network circumstances by analyzing incoming routing update messages. If the message indicates that a network change has occurred, the routing software recalculates routes and sends out new routing update messages. These messages permeate the network, stimulating routers to rerun their algorithms and change their routing tables accordingly.

Dynamic routing algorithms can be supplemented with static routes where appropriate. A router of last resort (a router to which all unroutable packets are sent), for example, can be designated to act as a repository for all unroutable packets, ensuring that all messages are at least handled in some way.

Note

There is no dynamic routing support in multi-context mode. As a result, there is no route tracking.

Single-Path Versus Multipath

Some sophisticated routing protocols support multiple paths to the same destination. Unlike single-path algorithms, these multipath algorithms permit traffic multiplexing over multiple lines. The advantages of multipath algorithms are substantially better throughput and reliability, which is generally called load sharing.

Flat Versus Hierarchical

Some routing algorithms operate in a flat space, while others use routing hierarchies. In a flat routing system, the routers are peers of all others. In a hierarchical routing system, some routers form what amounts to a routing backbone. Packets from nonbackbone routers travel to the backbone routers, where they are sent through the backbone until they reach the general area of the destination. At this point, they travel from the last backbone router through one or more nonbackbone routers to the final destination.

Routing systems often designate logical groups of nodes, called domains, autonomous systems, or areas. In hierarchical systems, some routers in a domain can communicate with routers in other domains, while others can communicate only with routers within their domain. In very large networks, additional hierarchical levels may exist, with routers at the highest hierarchical level forming the routing backbone.

The primary advantage of hierarchical routing is that it mimics the organization of most companies and therefore supports their traffic patterns well. Most network communication occurs within small company groups (domains). Because intradomain routers need to know only about other routers within their domain, their routing algorithms can be simplified, and, depending on the routing algorithm being used, routing update traffic can be reduced accordingly.
Link-State Versus Distance Vector

Link-state algorithms (also known as shortest path first algorithms) flood routing information to all nodes in the internetwork. Each router, however, sends only the portion of the routing table that describes the state of its own links. In link-state algorithms, each router builds a picture of the entire network in its routing tables. Distance vector algorithms (also known as Bellman-Ford algorithms) call for each router to send all or some portion of its routing table, but only to its neighbors. In essence, link-state algorithms send small updates everywhere, while distance vector algorithms send larger updates only to neighboring routers. Distance vector algorithms know only about their neighbors. Typically, this type of algorithm is used in conjunction with OSPF routing protocols.

How Routing Behaves Within the ASA

The ASASM uses both routing table and XLATE tables for routing decisions. To handle destination IP translated traffic, that is, untranslated traffic, the ASASM searches for existing XLATE, or static translation to select the egress interface.

This section includes the following topics:

- Egress Interface Selection Process, page 19-4
- Next Hop Selection Process, page 19-4

Egress Interface Selection Process

The selection process follows these steps:

1. If a destination IP translating XLATE already exists, the egress interface for the packet is determined from the XLATE table, but not from the routing table.
2. If a destination IP translating XLATE does not exist, but a matching static translation exists, then the egress interface is determined from the static route and an XLATE is created, and the routing table is not used.
3. If a destination IP translating XLATE does not exist and no matching static translation exists, the packet is not destination IP translated. The ASASM processes this packet by looking up the route to select the egress interface, then source IP translation is performed (if necessary).

For regular dynamic outbound NAT, initial outgoing packets are routed using the route table and then creating the XLATE. Incoming return packets are forwarded using existing XLATE only. For static NAT, destination translated incoming packets are always forwarded using existing XLATE or static translation rules.

Next Hop Selection Process

After selecting the egress interface using any method described previously, an additional route lookup is performed to find out suitable next hop(s) that belong to a previously selected egress interface. If there are no routes in the routing table that explicitly belong to a selected interface, the packet is dropped with
a level 6 syslog message 110001 generated (no route to host), even if there is another route for a given destination network that belongs to a different egress interface. If the route that belongs to a selected egress interface is found, the packet is forwarded to the corresponding next hop.

Load sharing on the ASASM is possible only for multiple next hops available using a single egress interface. Load sharing cannot share multiple egress interfaces.

If dynamic routing is in use on the ASASM and the route table changes after XLATE creation (for example, route flap), then destination translated traffic is still forwarded using the old XLATE, not via the route table, until XLATE times out. It may be either forwarded to the wrong interface or dropped with a level 6 syslog message 110001 generated (no route to host), if the old route was removed from the old interface and attached to another one by the routing process.

The same problem may happen when there are no route flaps on the ASASM itself, but some routing process is flapping around it, sending source-translated packets that belong to the same flow through the ASASM using different interfaces. Destination-translated return packets may be forwarded back using the wrong egress interface.

This issue has a high probability in some security traffic configurations, where virtually any traffic may be either source-translated or destination-translated, depending on the direction of the initial packet in the flow. When this issue occurs after a route flap, it can be resolved manually by using the `clear xlate` command, or automatically resolved by an XLATE timeout. The XLATE timeout may be decreased if necessary. To ensure that this issue rarely occurs, make sure that there are no route flaps on the ASASM and around it. That is, ensure that destination-translated packets that belong to the same flow are always forwarded the same way through the ASASM.

### Supported Internet Protocols for Routing

The ASASM supports several Internet protocols for routing. Each protocol is briefly described in this section.

- **Enhanced Interior Gateway Routing Protocol (EIGRP)**
  
  EIGRP provides compatibility and seamless interoperation with IGRP routers. An automatic-redistribution mechanism allows IGRP routes to be imported into Enhanced IGRP, and vice versa, so it is possible to add Enhanced IGRP gradually into an existing IGRP network.
  
  For more information about configuring EIGRP, see the “Configuring EIGRP” section on page 25-3.

- **Open Shortest Path First (OSPF)**
  
  Open Shortest Path First (OSPF) is a routing protocol developed for Internet Protocol (IP) networks by the interior gateway protocol (IGP) working group of the Internet Engineering Task Force (IETF). OSPF uses a link-state algorithm to build and calculate the shortest path to all known destinations. Each router in an OSPF area includes an identical link-state database, which is a list of each of the router usable interfaces and reachable neighbors.
  
  For more information about configuring OSPF, see the “Configuring OSPF” section on page 22-3.

- **Routing Information Protocol**
  
  The Routing Information Protocol (RIP) is a distance-vector protocol that uses hop count as its metric. RIP is widely used for routing traffic in the global Internet and is an interior gateway protocol (IGP), which means that it performs routing within a single autonomous system.
  
  For more information about configuring RIP, see the “Configuring RIP” section on page 23-4.
Information About the Routing Table

This section includes the following topics:

- Displaying the Routing Table, page 19-6
- How the Routing Table Is Populated, page 19-6
- How Forwarding Decisions Are Made, page 19-8
- Dynamic Routing and Failover, page 19-9

Displaying the Routing Table

To view the entries in the routing table, enter the following command:

```
hostname# show route
```

Codes:  
C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP  
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area  
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2  
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP  
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area  
* - candidate default, U - per-user static route, o - ODR  
P - periodic downloaded static route

Gateway of last resort is 10.86.194.1 to network 0.0.0.0

S  10.1.1.0 255.255.255.0 [3/0] via 10.86.194.1, outside
C  10.86.194.0 255.255.254.0 is directly connected, outside
S* 0.0.0.0 0.0.0.0 [1/0] via 10.86.194.1, outside

On the ASA 5505, the following route is also shown. It is the internal loopback interface, which is used by the VPN hardware client feature for individual user authentication.

```
C 127.1.0.0 255.255.0.0 is directly connected, _internal_loopback
```

How the Routing Table Is Populated

The ASASM routing table can be populated by statically defined routes, directly connected routes, and routes discovered by the RIP, EIGRP, and OSPF routing protocols. Because the ASASM can run multiple routing protocols in addition to having static and connected routes in the routing table, it is possible that the same route is discovered or entered in more than one manner. When two routes to the same destination are put into the routing table, the one that remains in the routing table is determined as follows:

- If the two routes have different network prefix lengths (network masks), then both routes are considered unique and are entered into the routing table. The packet forwarding logic then determines which of the two to use.

For example, if the RIP and OSPF processes discovered the following routes:

- RIP: 192.168.32.0/24
- OSPF: 192.168.32.0/19
Even though OSPF routes have the better administrative distance, both routes are installed in the routing table because each of these routes has a different prefix length (subnet mask). They are considered different destinations and the packet forwarding logic determines which route to use.

- If the ASASM learns about multiple paths to the same destination from a single routing protocol, such as RIP, the route with the better metric (as determined by the routing protocol) is entered into the routing table.

Metrics are values associated with specific routes, ranking them from most preferred to least preferred. The parameters used to determine the metrics differ for different routing protocols. The path with the lowest metric is selected as the optimal path and installed in the routing table. If there are multiple paths to the same destination with equal metrics, load balancing is done on these equal cost paths.

- If the ASASM learns about a destination from more than one routing protocol, the administrative distances of the routes are compared and the routes with lower administrative distance are entered into the routing table.

You can change the administrative distances for routes discovered by or redistributed into a routing protocol. If two routes from two different routing protocols have the same administrative distance, then the route with the lower default administrative distance is entered into the routing table. In the case of EIGRP and OSPF routes, if the EIGRP route and the OSPF route have the same administrative distance, then the EIGRP route is chosen by default.

Administrative distance is a route parameter that the ASASM uses to select the best path when there are two or more different routes to the same destination from two different routing protocols. Because the routing protocols have metrics based on algorithms that are different from the other protocols, it is not always possible to determine the best path for two routes to the same destination that were generated by different routing protocols.

Each routing protocol is prioritized using an administrative distance value. Table 19-1 shows the default administrative distance values for the routing protocols supported by the ASASM.

**Table 19-1 Default Administrative Distance for Supported Routing Protocols**

<table>
<thead>
<tr>
<th>Route Source</th>
<th>Default Administrative Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connected interface</td>
<td>0</td>
</tr>
<tr>
<td>Static route</td>
<td>1</td>
</tr>
<tr>
<td>EIGRP Summary Route</td>
<td>5</td>
</tr>
<tr>
<td>Internal EIGRP</td>
<td>90</td>
</tr>
<tr>
<td>OSPF</td>
<td>110</td>
</tr>
<tr>
<td>RIP</td>
<td>120</td>
</tr>
<tr>
<td>EIGRP external route</td>
<td>170</td>
</tr>
<tr>
<td>Unknown</td>
<td>255</td>
</tr>
</tbody>
</table>

The smaller the administrative distance value, the more preference is given to the protocol. For example, if the ASASM receives a route to a certain network from both an OSPF routing process (default administrative distance - 110) and a RIP routing process (default administrative distance - 120), the ASASM chooses the OSPF route because OSPF has a higher preference. In this case, the router adds the OSPF version of the route to the routing table.

In this example, if the source of the OSPF-derived route was lost (for example, due to a power shutdown), the ASASM would then use the RIP-derived route until the OSPF-derived route reappears.
The administrative distance is a local setting. For example, if you use the `distance-ospf` command to change the administrative distance of routes obtained through OSPF, that change would only affect the routing table for the ASASM on which the command was entered. The administrative distance is not advertised in routing updates.

Administrative distance does not affect the routing process. The OSPF and RIP routing processes only advertise the routes that have been discovered by the routing process or redistributed into the routing process. For example, the RIP routing process advertises RIP routes, even if routes discovered by the OSPF routing process are used in the ASASM routing table.

### Backup Routes

A backup route is registered when the initial attempt to install the route in the routing table fails because another route was installed instead. If the route that was installed in the routing table fails, the routing table maintenance process calls each routing protocol process that has registered a backup route and requests them to reinstall the route in the routing table. If there are multiple protocols with registered backup routes for the failed route, the preferred route is chosen based on administrative distance.

Because of this process, you can create floating static routes that are installed in the routing table when the route discovered by a dynamic routing protocol fails. A floating static route is simply a static route configured with a greater administrative distance than the dynamic routing protocols running on the ASASM. When the corresponding route discovered by a dynamic routing process fails, the static route is installed in the routing table.

### How Forwarding Decisions Are Made

Forwarding decisions are made as follows:

- If the destination does not match an entry in the routing table, the packet is forwarded through the interface specified for the default route. If a default route has not been configured, the packet is discarded.
- If the destination matches a single entry in the routing table, the packet is forwarded through the interface associated with that route.
- If the destination matches more than one entry in the routing table, and the entries all have the same network prefix length, the packets for that destination are distributed among the interfaces associated with that route.
- If the destination matches more than one entry in the routing table, and the entries have different network prefix lengths, then the packet is forwarded out of the interface associated with the route that has the longer network prefix length.

For example, a packet destined for 192.168.32.1 arrives on an interface of an ASASM with the following routes in the routing table:

```
hostname# show route
....
R  192.168.32.0/24 [120/4] via 10.1.1.2
O  192.168.32.0/19 [110/229840] via 10.1.1.3
....
```

In this case, a packet destined to 192.168.32.1 is directed toward 10.1.1.2, because 192.168.32.1 falls within the 192.168.32.0/24 network. It also falls within the other route in the routing table, but the 192.168.32.0/24 has the longest prefix within the routing table (24 bits verses 19 bits). Longer prefixes are always preferred over shorter ones when forwarding a packet.
Dynamic Routing and Failover

Because static routing systems cannot react to network changes, they generally are considered unsuitable for large, constantly changing networks. Most of the dominant routing algorithms are dynamic routing algorithms, which adjust to changing network circumstances by analyzing incoming routing update messages. If the message indicates that a network change has occurred, the routing software recalculates routes and sends out new routing update messages. These messages permeate the network, stimulating routers to rerun their algorithms and change their routing tables accordingly.

Dynamic routing algorithms can be supplemented with static routes where appropriate. A router of last resort (a router to which all unroutable packets are sent), for example, can be designated to act as a repository for all unroutable packets, ensuring that all messages are at least handled in some way.

Dynamic routes are synchronized on the standby unit when the routing table changes on the active unit, which means that all additions, deletions, or changes on the active unit are immediately propagated to the standby unit. If the standby unit becomes active after the primary unit has been active for a period of time, routes become synchronized as a part of the failover bulk synchronization process, so the routing table on the active/standby failover pair should appear the same.

For more information about static routes and how to configure them, see the “Configuring Static and Default Routes” section on page 20-2.

Information About IPv6 Support

Many, but not all, features on the ASASM support IPv6 traffic. This section describes the commands and features that support IPv6 and includes the following topics:

- Features That Support IPv6, page 19-9
- IPv6-Enabled Commands, page 19-10
- Entering IPv6 Addresses in Commands, page 19-11

Features That Support IPv6

The following features support IPv6:

<table>
<thead>
<tr>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>For features that use the Modular Policy Framework, be sure to use the \texttt{match any} command to match IPv6 traffic; other \texttt{match} commands do not support IPv6.</td>
</tr>
</tbody>
</table>

- The following application inspections support IPv6 traffic:
  - FTP
  - HTTP
  - ICMP
  - SIP
  - SMTP
  - IPsec-pass-thru
- IPS
Information About IPv6 Support

- NetFlow Secure Event Logging filtering
- Connection limits, timeouts, and TCP randomization
- TCP Normalization
- TCP state bypass
- Access group, using an IPv6 access list
- Static Routes
- VPN (all types)
- Failover
- Transparent firewall mode

IPv6-Enabled Commands

The following ASASM commands can accept and display IPv6 addresses:

- capture
- configure
- copy
- failover interface ip
- http
- name
- object-group
- ping
- show conn
- show local-host
- show tcpstat
- ssh
- telnet
- tftp-server
- who
- write

The following commands were modified to work for IPv6:

- debug
- fragment
- ip verify
- mtu
- icmp (entered as ipv6 icmp)
Entering IPv6 Addresses in Commands

When entering IPv6 addresses in commands that support them, enter the IPv6 address using standard IPv6 notation, for example:

```
ping fe80::2e0:b6ff:fe01:3b7a.
```

The ASASM correctly recognizes and processes the IPv6 address. However, you must enclose the IPv6 address in square brackets ([ ]) in the following situations:

- You need to specify a port number with the address, for example:
  
  `[fe80::2e0:b6ff:fe01:3b7a]:8080`.

- The command uses a colon as a separator, such as the `write net` command and `config net` command, for example:

  `configure net [fe80::2e0:b6ff:fe01:3b7a]:tftp/config/asaconfig`.

Disabling Proxy ARPs

When a host sends IP traffic to another device on the same Ethernet network, the host needs to know the MAC address of the device. ARP is a Layer 2 protocol that resolves an IP address to a MAC address. A host sends an ARP request asking “Who is this IP address?” The device owning the IP address replies, “I own that IP address; here is my MAC address.”

Proxy ARP is used when a device responds to an ARP request with its own MAC address, even though the device does not own the IP address. The ASASM uses proxy ARP when you configure NAT and specify a mapped address that is on the same network as the ASASM interface. The only way traffic can reach the hosts is if the ASASM uses proxy ARP to claim that the MAC address is assigned to destination mapped addresses.

Under rare circumstances, you might want to disable proxy ARP for NAT addresses.

If you have a VPN client address pool that overlaps with an existing network, the ASASM by default sends proxy ARPs on all interfaces. If you have another interface that is on the same Layer 2 domain, it will see the ARP requests and will answer with the MAC address of its interface. The result of this is that the return traffic of the VPN clients towards the internal hosts will go to the wrong interface and will get dropped. In this case, you need to disable proxy ARPs for the interface on which you do not want proxy ARPs.

To disable proxy ARPs, enter the following command:

```
Command | Purpose
--------|--------
sysopt noproxyarp interface | Disables proxy ARPs.
```

Example:
```
hostname(config)# sysopt noproxyarp exampleinterface
```
CHAPTER 20
Configuring Static and Default Routes

This chapter describes how to configure static and default routes on the ASASM and includes the following sections:

- Information About Static and Default Routes, page 20-1
- Licensing Requirements for Static and Default Routes, page 20-2
- Guidelines and Limitations, page 20-2
- Configuring Static and Default Routes, page 20-2
- Monitoring a Static or Default Route, page 20-6
- Configuration Examples for Static or Default Routes, page 20-8
- Feature History for Static and Default Routes, page 20-8

**Information About Static and Default Routes**

To route traffic to a nonconnected host or network, you must define a static route to the host or network or, at a minimum, a default route for any networks to which the ASASM is not directly connected; for example, when there is a router between a network and the ASASM.

Without a static or default route defined, traffic to nonconnected hosts or networks generates the following syslog message:

`%ASA-6-110001: No route to dest_address from source_address`

Multiple context mode does not support dynamic routing,
You might want to use static routes in single context mode in the following cases:

- Your networks use a different router discovery protocol from EIGRP, RIP, or OSPF.
- Your network is small and you can easily manage static routes.
- You do not want the traffic or CPU overhead associated with routing protocols.

The simplest option is to configure a default route to send all traffic to an upstream router, relying on the router to route the traffic for you. However, in some cases the default gateway might not be able to reach the destination network, so you must also configure more specific static routes. For example, if the default gateway is outside, then the default route cannot direct traffic to any inside networks that are not directly connected to the ASASM.

In transparent firewall mode, for traffic that originates on the ASASM and is destined for a nondirectly connected network, you need to configure either a default route or static routes so the ASASM knows out of which interface to send traffic. Traffic that originates on the ASASM might include
communications to a syslog server, Websense or N2H2 server, or AAA server. If you have servers that cannot all be reached through a single default route, then you must configure static routes. Additionally, the ASASM supports up to three equal cost routes on the same interface for load balancing.

### Licensing Requirements for Static and Default Routes

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

### Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

**Context Mode Guidelines**
Supported in single and multiple context mode.

**Firewall Mode Guidelines**
Supported in routed and transparent firewall mode.

**IPv6 Guidelines**
Supports IPv6.

**Failover Guidelines**
Supports stateful failover of dynamic routing protocols.

**Additional Guidelines**
IPv6 static routes are not supported in transparent mode in ASDM.

### Configuring Static and Default Routes

This section explains how to configure a static route and a static default route, and includes the following topics:

- Configuring a Static Route, page 20-3
- Configuring a Default Static Route, page 20-4
- Configuring IPv6 Default and Static Routes, page 20-5
Configuring a Static Route

Static routing algorithms are basically table mappings established by the network administrator before the beginning of routing. These mappings do not change unless the network administrator alters them. Algorithms that use static routes are simple to design and work well in environments where network traffic is relatively predictable and where network design is relatively simple. Because of this fact, static routing systems cannot react to network changes.

Static routes remain in the routing table even if the specified gateway becomes unavailable. If the specified gateway becomes unavailable, you need to remove the static route from the routing table manually. However, static routes are removed from the routing table if the specified interface goes down, and are reinstated when the interface comes back up.

Note

If you create a static route with an administrative distance greater than the administrative distance of the routing protocol running on the ASASM, then a route to the specified destination discovered by the routing protocol takes precedence over the static route. The static route is used only if the dynamically discovered route is removed from the routing table.

You can define up to three equal cost routes to the same destination per interface. Equal-cost multi-path (ECMP) routing is not supported across multiple interfaces. With ECMP, the traffic is not necessarily divided evenly between the routes; traffic is distributed among the specified gateways based on an algorithm that hashes the source and destination IP addresses.

To configure a static route, see the following section:

- Adding or Editing a Static Route, page 20-3

Adding or Editing a Static Route

To add or edit a static route, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>route if_name dest_ip mask gateway_ip [distance]</code></td>
<td>Enables you to add a static route. The <code>dest_ip</code> and <code>mask</code> arguments indicate the IP address for the destination network, and the <code>gateway_ip</code> argument is the address of the next-hop router. The addresses you specify for the static route are the addresses that are in the packet before entering the ASASM and performing NAT. The <code>distance</code> argument is the administrative distance for the route. The default is 1 if you do not specify a value. Administrative distance is a parameter used to compare routes among different routing protocols. The default administrative distance for static routes is 1, giving it precedence over routes discovered by dynamic routing protocols but not directly connected routes. The default administrative distance for routes discovered by OSPF is 110. If a static route has the same administrative distance as a dynamic route, the static route takes precedence. Connected routes always take precedence over static or dynamically discovered routes.</td>
</tr>
</tbody>
</table>

Example:

```
hostname(config)# route outside 10.10.10.0 255.255.255.0 192.168.1.1 [1]
```
Examples

The following example shows static routes that are equal cost routes that direct traffic to three different gateways on the outside interface. The ASASM distributes the traffic among the specified gateways.

```
hostname(config)# route outside 10.10.10.0 255.255.255.0 192.168.1.1
hostname(config)# route outside 10.10.10.0 255.255.255.0 192.168.1.2
hostname(config)# route outside 10.10.10.0 255.255.255.0 192.168.1.3
```

Configuring a Default Static Route

A default route identifies the gateway IP address to which the ASASM sends all IP packets for which it does not have a learned or static route. A default static route is simply a static route with 0.0.0.0/0 as the destination IP address. Routes that identify a specific destination take precedence over the default route.

**Note**

In Versions 7.0(1) and later, if you have two default routes configured on different interfaces that have different metrics, the connection to the ASASM that is made from the higher metric interface fails, but connections to the ASASM from the lower metric interface succeed as expected.

You can define up to three equal cost default route entries per device. Defining more than one equal cost default route entry causes the traffic sent to the default route to be distributed among the specified gateways. When defining more than one default route, you must specify the same interface for each entry.

If you attempt to define more than three equal cost default routes or a default route with a different interface than a previously defined default route, you receive the following message:

```
"ERROR: Cannot add route entry, possible conflict with existing routes."
```

You can define a separate default route for tunneled traffic along with the standard default route. When you create a default route with the tunneled option, all traffic from a tunnel terminating on the ASASM that cannot be routed using learned or static routes is sent to this route. For traffic emerging from a tunnel, this route overrides any other configured or learned default routes.

Limitations on Configuring a Default Static Route

The following restrictions apply to default routes with the tunneled option:

- Do not enable unicast RPF (**ip verify reverse-path** command) on the egress interface of a tunneled route, because this setting causes the session to fail.
- Do not enable TCP intercept on the egress interface of the tunneled route, because this setting causes the session to fail.
- Do not use the VoIP inspection engines (CTIQBE, H.323, GTP, MGCP, RTSP, SIP, SKINNY), the DNS inspect engine, or the DCE RPC inspection engine with tunneled routes, because these inspection engines ignore the tunneled route.
- You cannot define more than one default route with the tunneled option.
- ECMP for tunneled traffic is not supported.

To add or edit a tunneled default static route, enter the following command:
Chapter 20 Configuring Static and Default Routes

### Configuring Static and Default Routes

Tip

You can enter 0 0 instead of 0.0.0.0 0.0.0.0 for the destination network address and mask, as shown in the following example:

```
hostname(config)# route outside 0 0 192.168.2.4 tunneled
```

### Configuring IPv6 Default and Static Routes

The ASASM automatically routes IPv6 traffic between directly connected hosts if the interfaces to which the hosts are attached are enabled for IPv6 and the IPv6 ACLs allow the traffic.

To configure an IPv6 default route and static routes, perform the following steps:

#### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><strong>Step 2</strong></td>
</tr>
<tr>
<td><code>ipv6 route if_name ::/0 next_hop_ipv6_addr</code></td>
<td>Adds an IPv6 static route to the IPv6 routing table.</td>
</tr>
<tr>
<td></td>
<td>The example routes packets for network 7fff::0/32 to a networking device on the inside interface at 3FFE:1100:0:CC00::1, and with an administrative distance of 110.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td><code>hostname(config)# ipv6 route inside 7fff::0/32 3FFE:1100:0:CC00::1</code></td>
<td><code>hostname(config)# ipv6 route inside 7fff::0/32 3FFE:1100:0:CC00::1 [110]</code></td>
</tr>
</tbody>
</table>

The example routes packets for network 7fff::0/32 to a networking device on the inside interface at 3FFE:1100:0:CC00::1, and with an administrative distance of 110.
Monitoring a Static or Default Route

One of the problems with static routes is that there is no inherent mechanism for determining if the route is up or down. They remain in the routing table even if the next hop gateway becomes unavailable. Static routes are only removed from the routing table if the associated interface on the ASASM goes down.

The static route tracking feature provides a method for tracking the availability of a static route and installing a backup route if the primary route should fail. For example, you can define a default route to an ISP gateway and a backup default route to a secondary ISP in case the primary ISP becomes unavailable.

The ASASM implements this feature by associating a static route with a monitoring target that you define, and monitors the target using ICMP echo requests. If an echo reply is not received within a specified time period, the object is considered down and the associated route is removed from the routing table. A previously configured backup route is used in place of the removed route.

When selecting a monitoring target, you need to make sure that it can respond to ICMP echo requests. The target can be any network object that you choose, but you should consider using the following:

- The ISP gateway (for dual ISP support) address
- The next hop gateway address (if you are concerned about the availability of the gateway)
- A server on the target network, such as a AAA server, that the ASASM needs to communicate with
- A persistent network object on the destination network

A desktop or notebook computer that may be shut down at night is not a good choice.

You can configure static route tracking for statically defined routes or default routes obtained through DHCP or PPPoE. You can only enable PPPoE clients on multiple interfaces with route tracking configured.

To configure static route tracking, perform the following steps:
### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> sla monitor sla_id</td>
<td>Configures the tracked object monitoring parameters by defining the monitoring process. If you are configuring a new monitoring process, you enter sla monitor configuration mode. If you are changing the monitoring parameters for an unscheduled monitoring process that already has a type defined, you automatically enter sla protocol configuration mode.</td>
</tr>
</tbody>
</table>

**Example:**
```
hostname(config)# sla monitor sla_id
```

| **Step 2** type echo protocol ipIcmpEcho target_ip interface if_name | Specifies the monitoring protocol. If you are changing the monitoring parameters for an unscheduled monitoring process that already has a type defined, you automatically enter sla protocol configuration mode and cannot change this setting. The `target_ip` argument is the IP address of the network object whose availability the tracking process monitors. While this object is available, the tracking process route is installed in the routing table. When this object becomes unavailable, the tracking process removes the route and the backup route is used in its place. |

**Example:**
```
hostname(config-sla-monitor)# type echo protocol ipIcmpEcho target_ip interface if_name
```

| **Step 3** sla monitor schedule sla_id [life {forever | seconds}] [start-time {hh:mm:ss [month day | day month] | pending | now | after hh:mm:ss}] [ageout seconds] [recurring] | Schedules the monitoring process. Typically, you will use the `sla monitor schedule sla_id life forever start-time now` command for the monitoring schedule, and allow the monitoring configuration to determine how often the testing occurs. However, you can schedule this monitoring process to begin in the future and to only occur at specified times. |

**Example:**
```
hostname(config)# sla monitor schedule sla_id [life {forever | seconds}] [start-time {hh:mm:ss [month day | day month] | pending | now | after hh:mm:ss}] [ageout seconds] [recurring]
```

| **Step 4** track track_id rtr sla_id reachability | Associates a tracked static route with the SLA monitoring process. The `track_id` argument is a tracking number you assign with this command. The `sla_id` argument is the ID number of the SLA process. |

**Example:**
```
hostname(config)# track track_id rtr sla_id reachability
```

| **Step 5** | Do one of the following to define the static route to be installed in the routing table while the tracked object is reachable. These options allow you to track a static route or a default route obtained through DHCP or PPPOE. Tracks a static route. You cannot use the `tunneled` option with the `route` command in static route tracking. |

| route if_name dest_ip mask gateway_ip [admin_distance] track track_id | |

**Example:**
```
hostname(config)# route if_name dest_ip mask gateway_ip [admin_distance] track track_id
```

---

*Cisco ASA Services Module CLI Configuration Guide*
Configuration Examples for Static or Default Routes

The following example shows how to create a static route that sends all traffic destined for 10.1.1.0/24 to the router 10.1.2.45, which is connected to the inside interface, defines three equal cost static routes that direct traffic to three different gateways on the outside interface, and adds a default route for tunneled traffic. The ASASM then distributes the traffic among the specified gateways:

```
hostname(config)# route inside 10.1.1.0 255.255.255.0 10.1.2.45 1
hostname(config)# route outside 10.10.10.0 255.255.255.0 192.168.2.1
hostname(config)# route outside 10.10.10.0 255.255.255.0 192.168.2.2
hostname(config)# route outside 10.10.10.0 255.255.255.0 192.168.2.3
hostname(config)# route outside 0 0 192.168.2.4 tunneled
```

Unencrypted traffic received by the ASASM for which there is no static or learned route is distributed among the gateways with the IP addresses 192.168.2.1, 192.168.2.2, and 192.168.2.3. Encrypted traffic received by the ASASM for which there is no static or learned route is passed to the gateway with the IP address 192.168.2.4.

The following example creates a static route that sends all traffic destined for 10.1.1.0/24 to the router (10.1.2.45) connected to the inside interface:

```
hostname(config)# route inside 10.1.1.0 255.255.255.0 10.1.2.45 1
```

Feature History for Static and Default Routes

Table 20-1 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Routing</td>
<td>7.0(1)</td>
<td>Static and default routing were introduced. We introduced the <code>route</code> command.</td>
</tr>
</tbody>
</table>
Defining Route Maps

This chapter describes route maps and includes the following sections:

- Information About Route Maps, page 21-1
- Licensing Requirements for Route Maps, page 21-3
- Guidelines and Limitations, page 21-3
- Defining a Route Map, page 21-4
- Customizing a Route Map, page 21-4
- Configuration Example for Route Maps, page 21-6
- Feature History for Route Maps, page 21-6

Information About Route Maps

Route maps are used when redistributing routes into an OSPF, RIP, or EIGRP routing process. They are also used when generating a default route into an OSPF routing process. A route map defines which of the routes from the specified routing protocol are allowed to be redistributed into the target routing process.

Route maps have many features in common with widely known ACLs. These are some of the traits common to both:

- They are an ordered sequence of individual statements, each has a permit or deny result. Evaluation of ACL or route maps consists of a list scan, in a predetermined order, and an evaluation of the criteria of each statement that matches. A list scan is aborted once the first statement match is found and an action associated with the statement match is performed.
- They are generic mechanisms—Criteria matches and match interpretation are dictated by the way that they are applied. The same route map applied to different tasks might be interpreted differently.

These are some of the differences between route maps and ACLs:

- Route maps frequently use ACLs as matching criteria.
- The main result from the evaluation of an access list is a yes or no answer—An ACL either permits or denies input data. Applied to redistribution, an ACL determines if a particular route can (route matches ACLs permit statement) or can not (matches deny statement) be redistributed. Typical route maps not only permit (some) redistributed routes but also modify information associated with the route, when it is redistributed into another protocol.
- Route maps are more flexible than ACLs and can verify routes based on criteria which ACLs can not verify. For example, a route map can verify if the type of route is internal.
Each ACL ends with an implicit deny statement, by design convention; there is no similar convention for route maps. If the end of a route map is reached during matching attempts, the result depends on the specific application of the route map. Fortunately, route maps that are applied to redistribution behave the same way as ACLs: if the route does not match any clause in a route map then the route redistribution is denied, as if the route map contained deny statement at the end.

The dynamic protocol `redistribute` command allows you to apply a route map. In ASDM, this capability for redistribution can be found when you add or edit a new route map (see the “Defining a Route Map” section on page 21-4). Route maps are preferred if you intend to either modify route information during redistribution or if you need more powerful matching capability than an ACL can provide. If you simply need to selectively permit some routes based on their prefix or mask, we recommend that you use a route map to map to an ACL (or equivalent prefix list) directly in the `redistribute` command. If you use a route map to selectively permit some routes based on their prefix or mask, you typically use more configuration commands to achieve the same goal.

**Note**

You must use a standard ACL as the match criterion for your route map. Using an extended ACL will not work, and your routes will never be redistributed. We recommend that you number clauses in intervals of 10, to reserve numbering space in case you need to insert clauses in the future.

This section includes the following topics:

- Permit and Deny Clauses, page 21-2
- Match and Set Clause Values, page 21-2

### Permit and Deny Clauses

Route maps can have permit and deny clauses. In the `route-map ospf-to-eigrp` command, there is one deny clause (with sequence number 10) and two permit clauses. The deny clause rejects route matches from redistribution. Therefore, the following rules apply:

- If you use an ACL in a route map using a permit clause, routes that are permitted by the ACL are redistributed.
- If you use an ACL in a route map deny clause, routes that are permitted by the ACL are not redistributed.
- If you use an ACL in a route map permit or deny clause, and the ACL denies a route, then the route map clause match is not found and the next route-map clause is evaluated.

### Match and Set Clause Values

Each route map clause has two types of values:

- A match value selects routes to which this clause should be applied.
- A set value modifies information that will be redistributed into the target protocol.

For each route that is being redistributed, the router first evaluates the match criteria of a clause in the route map. If the match criteria succeed, then the route is redistributed or rejected as dictated by the permit or deny clause, and some of its attributes might be modified by the values set from the Set Value tab in ASDM or from the `set` commands. If the match criteria fail, then this clause is not applicable to the route, and the software proceeds to evaluate the route against the next clause in the route map. Scanning of the route map continues until a clause is found whose `match` command(s), or Match Clause as set from the Match Clause tab in ASDM, match the route or until the end of the route map is reached.
A match or set value in each clause can be missed or repeated several times, if one of these conditions exists:

- If several **match** commands or Match Clause values in ASDM are present in a clause, all must succeed for a given route in order for that route to match the clause (in other words, the logical AND algorithm is applied for multiple match commands).

- If a **match** command or Match Clause value in ASDM refers to several objects in one command, either of them should match (the logical OR algorithm is applied). For example, in the **match ip address 101 121** command, a route is permitted if access list 101 or access list 121 permits it.

- If a **match** command or Match Clause value in ASDM is not present, all routes match the clause. In the previous example, all routes that reach clause 30 match; therefore, the end of the route map is never reached.

- If a **set** command, or Set Value in ASDM, is not present in a route map permit clause, then the route is redistributed without modification of its current attributes.

---

**Note**

Do not configure a **set** command in a route map deny clause because the deny clause prohibits route redistribution—there is no information to modify.

---

A route map clause without a **match** or **set** command, or Match or Set Value as set on the Match or Set Value tab in ASDM, performs an action. An empty permit clause allows a redistribution of the remaining routes without modification. An empty deny clause does not allows a redistribution of other routes (this is the default action if a route map is completely scanned, but no explicit match is found).

---

**Licensing Requirements for Route Maps**

The following table shows the licensing requirements for route maps:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License</td>
</tr>
</tbody>
</table>

---

**Guidelines and Limitations**

This section includes the guidelines and limitations for this feature.

**Context Mode Guidelines**

Supported in single context mode.

**Firewall Mode Guidelines**

Supported only in routed firewall mode. Transparent firewall mode is not supported.

**IPv6 Guidelines**

Does not support IPv6.
Defining a Route Map

You must define a route map when specifying which of the routes from the specified routing protocol are allowed to be redistributed into the target routing process.

To define a route map, enter the following command:

```
route-map name (permit | deny) [sequence_number]
```

Example:

```
hostname(config)# route-map name (permit) [12]
```

Customizing a Route Map

This section describes how to customize the route map and includes the following topics:

- Defining a Route to Match a Specific Destination Address, page 21-4
- Configuring the Metric Values for a Route Action, page 21-5

Defining a Route to Match a Specific Destination Address

To define a route to match a specified destination address, perform the following steps:

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`route-map name (permit</td>
<td>deny) [sequence_number]`</td>
</tr>
<tr>
<td><code>match ip address acl_id acl_id [...] [prefix-list]</code></td>
<td>Matches any routes that have a destination network that matches a standard ACL or prefix list. If you specify more than one ACL, then the route can match any of the ACLs.</td>
</tr>
</tbody>
</table>

Example:

```
hostname(config-route-map)# match ip address acl_id acl_id [...] [prefix-list]
```
### Customizing a Route Map

#### Configuring the Metric Values for a Route Action

If a route matches the **match** commands, then the following **set** commands determine the action to perform on the route before redistributing it.

To configure the metric value for a route action, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>match metric metric_value</code></td>
<td>Matches any routes that have a specified metric. The <em>metric_value</em> can range from 0 to 4294967295.</td>
</tr>
</tbody>
</table>

**Example:**
```
hostname(config-route-map)# match metric 200
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>match ip next-hop acl_id [acl_id] [...]</code></td>
<td>Matches any routes that have a next hop router address that matches a standard ACL. If you specify more than one ACL, then the route can match any of the ACLs.</td>
</tr>
</tbody>
</table>

**Example:**
```
hostname(config-route-map)# match ip next-hop acl_id [acl_id] [...] 
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>match interface if_name</code></td>
<td>Matches any routes with the specified next hop interface. If you specify more than one interface, then the route can match either interface.</td>
</tr>
</tbody>
</table>

**Example:**
```
hostname(config-route-map)# match interface if_name
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>match ip route-source acl_id [acl_id] [...]</code></td>
<td>Matches any routes that have been advertised by routers that match a standard ACL. If you specify more than one ACL, then the route can match any of the ACLs.</td>
</tr>
</tbody>
</table>

**Example:**
```
hostname(config-route-map)# match ip route-source acl_id [acl_id] [...] 
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`match route-type (internal</td>
<td>external [type-1</td>
</tr>
</tbody>
</table>

**Example:**
```
hostname(config-route-map)# match route-type internal type-1
```

---

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`route-map name (permit</td>
<td>deny) [sequence_number]`</td>
</tr>
</tbody>
</table>

**Example:**
```
hostname(config)# route-map name {permit} [12]
```

**Step 2**

To set a metric for the route map, enter one or more of the following **set** commands:
The following example shows how to redistribute routes with a hop count equal to 1 into OSPF.

The ASASM redistributes these routes as external LSAs with a metric of 5 and a metric type of Type 1.

```
hostname(config)# route-map 1-to-2 permit
hostname(config-route-map)# match metric 1
hostname(config-route-map)# set metric 5
hostname(config-route-map)# set metric-type type-1
```

The following example shows how to redistribute the 10.1.1.0 static route into eigrp process 1 with the configured metric value:

```
hostname(config)# route outside 10.1.1.0 255.255.255.0 192.168.1.1
hostname(config-route-map)# access-list mymap2 line 1 permit 10.1.1.0 255.255.255.0
hostname(config-route-map)# route-map mymap2 permit 10
hostname(config-route-map)# match ip address mymap2
hostname(config-route-map)# router eigrp 1
hostname(config)# redistribute static metric 250 250 1 1 1 route-map mymap2
```

### Feature History for Route Maps

Table 21-1 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Route maps</td>
<td>7.0(1)</td>
<td>We introduced this feature. We introduced the following command: route-map.</td>
</tr>
<tr>
<td>Enhanced support for static and dynamic route maps</td>
<td>8.0(2)</td>
<td>Enhanced support for dynamic and static route maps was added.</td>
</tr>
<tr>
<td>Support for stateful failover of dynamic routing protocols (EIGRP, OSPF, and RIP) and debugging of general routing-related operations</td>
<td>8.4(1)</td>
<td>We introduced the following commands: debug route, show debug route. We modified the following command: show route.</td>
</tr>
</tbody>
</table>
Configuring OSPF

This chapter describes how to configure the ASASM to route data, perform authentication, and redistribute routing information using the Open Shortest Path First (OSPF) routing protocol.

The chapter includes the following sections:

- Information About OSPF, page 22-1
- Licensing Requirements for OSPF, page 22-2
- Guidelines and Limitations, page 22-3
- Configuring OSPF, page 22-3
- Customizing OSPF, page 22-4
- Restarting the OSPF Process, page 22-14
- Configuration Example for OSPF, page 22-14
- Monitoring OSPF, page 22-16
- Feature History for OSPF, page 22-17

Information About OSPF

OSPF is an interior gateway routing protocol that uses link states rather than distance vectors for path selection. OSPF propagates link-state advertisements rather than routing table updates. Because only LSAs are exchanged instead of the entire routing tables, OSPF networks converge more quickly than RIP networks.

OSPF uses a link-state algorithm to build and calculate the shortest path to all known destinations. Each router in an OSPF area contains an identical link-state database, which is a list of each of the router usable interfaces and reachable neighbors.

The advantages of OSPF over RIP include the following:

- OSPF link-state database updates are sent less frequently than RIP updates, and the link-state database is updated instantly, rather than gradually, as stale information is timed out.
- Routing decisions are based on cost, which is an indication of the overhead required to send packets across a certain interface. The ASASM calculates the cost of an interface based on link bandwidth rather than the number of hops to the destination. The cost can be configured to specify preferred paths.

The disadvantage of shortest path first algorithms is that they require a lot of CPU cycles and memory.
The ASASM can run two processes of OSPF protocol simultaneously on different sets of interfaces. You might want to run two processes if you have interfaces that use the same IP addresses (NAT allows these interfaces to coexist, but OSPF does not allow overlapping addresses). Or you might want to run one process on the inside and another on the outside, and redistribute a subset of routes between the two processes. Similarly, you might need to segregate private addresses from public addresses.

You can redistribute routes into an OSPF routing process from another OSPF routing process, a RIP routing process, or from static and connected routes configured on OSPF-enabled interfaces.

The ASASM supports the following OSPF features:

- Support of intra-area, interarea, and external (Type I and Type II) routes.
- Support of a virtual link.
- OSPF LSA flooding.
- Authentication to OSPF packets (both password and MD5 authentication).
- Support for configuring the ASASM as a designated router or a designated backup router. The ASASM also can be set up as an ABR.
- Support for stub areas and not-so-stubby areas.
- Area boundary router Type 3 LSA filtering.

OSPF supports MD5 and clear text neighbor authentication. Authentication should be used with all routing protocols when possible because route redistribution between OSPF and other protocols (like RIP) can potentially be used by attackers to subvert routing information.

If NAT is used, if OSPF is operating on public and private areas, and if address filtering is required, then you need to run two OSPF processes—one process for the public areas and one for the private areas.

A router that has interfaces in multiple areas is called an Area Border Router (ABR). A router that acts as a gateway to redistribute traffic between routers using OSPF and routers using other routing protocols is called an Autonomous System Boundary Router (ASBR).

An ABR uses LSAs to send information about available routes to other OSPF routers. Using ABR Type 3 LSA filtering, you can have separate private and public areas with the ASASM acting as an ABR. Type 3 LSAs (interarea routes) can be filtered from one area to other, which allows you to use NAT and OSPF together without advertising private networks.

**Note**

Only Type 3 LSAs can be filtered. If you configure the ASASM as an ASBR in a private network, it will send Type 5 LSAs describing private networks, which will get flooded to the entire AS, including public areas.

If NAT is employed but OSPF is only running in public areas, then routes to public networks can be redistributed inside the private network, either as default or Type 5 AS External LSAs. However, you need to configure static routes for the private networks protected by the ASASM. Also, you should not mix public and private networks on the same ASASM interface.

You can have two OSPF routing processes, one RIP routing process, and one EIGRP routing process running on the ASASM at the same time.

## Licensing Requirements for OSPF

The following table shows the licensing requirements for this feature:
Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

**Context Mode Guidelines**
Supported in single context mode.

**Firewall Mode Guidelines**
Supported in routed firewall mode only. Transparent firewall mode is not supported.

**IPv6 Guidelines**
Does not support IPv6.

Configuring OSPF

This section describes how to enable an OSPF process on the ASASM.

After you enable OSPF, you need to define a route map. For more information, see the “Defining a Route Map” section on page 21-4. Then you generate a default route. For more information, see the “Configuring Static and Default Routes” section on page 20-2.

After you have defined a route map for the OSPF process, you can customize the OSPF process to suit your particular needs. To learn how to customize the OSPF process on the ASASM, see the “Customizing OSPF” section on page 22-4.

To enable OSPF, you need to create an OSPF routing process, specify the range of IP addresses associated with the routing process, then assign area IDs associated with that range of IP addresses.

You can enable up to two OSPF process instances. Each OSPF process has its own associated areas and networks.

To enable OSPF, perform the following steps:
Customizing OSPF

This section explains how to customize the OSPF process and includes the following topics:

- Redistributing Routes Into OSPF, page 22-4
- Configuring Route Summarization When Redistributing Routes Into OSPF, page 22-6
- Configuring Route Summarization Between OSPF Areas, page 22-7
- Configuring OSPF Interface Parameters, page 22-8
- Configuring OSPF Area Parameters, page 22-10
- Configuring OSPF NSSA, page 22-11
- Defining Static OSPF Neighbors, page 22-12
- Configuring Route Calculation Timers, page 22-13
- Logging Neighbors Going Up or Down, page 22-13

Redistributing Routes Into OSPF

The ASASM can control the redistribution of routes between OSPF routing processes.

Note

If you want to redistribute a route by defining which of the routes from the specified routing protocol are allowed to be redistributed into the target routing process, you must first generate a default route. See the “Configuring Static and Default Routes” section on page 20-2, and then define a route map according to the “Defining a Route Map” section on page 21-4.

To redistribute static, connected, RIP, or OSPF routes into an OSPF process, perform the following steps:
## Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><strong>router ospf process_id</strong>&lt;br&gt;<strong>Example:</strong> hostname(config)# router ospf 2</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Do one of the following to redistribute the selected route type into the OSPF routing process:&lt;br&gt;&lt;br&gt;<strong>redistribute connected</strong>&lt;br&gt;[[metric metric-value] [metric-type {type-1</td>
</tr>
<tr>
<td></td>
<td><strong>redistribute static</strong> [metric metric-value] [metric-type {type-1</td>
</tr>
<tr>
<td></td>
<td><strong>redistribute ospf pid</strong> [match {internal</td>
</tr>
</tbody>
</table>
### Customizing OSPF

#### Configuring Route Summarization When Redistributing Routes Into OSPF

When routes from other protocols are redistributed into OSPF, each route is advertised individually in an external LSA. However, you can configure the ASASM to advertise a single route for all the redistributed routes that are included for a specified network address and mask. This configuration decreases the size of the OSPF link-state database.

Routes that match the specified IP Address mask pair can be suppressed. The tag value can be used as a match value for controlling redistribution through route maps.

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>redistribute rip [metric metric-value] [metric-type {type-1</td>
<td>type-2}] [tag tag_value] [subnets] [route-map map_name]</td>
</tr>
<tr>
<td><strong>Example:</strong> hostname(config)# redistribute rip 5 hostname(config-route-map)# match metric 1 hostname(config-route-map)# set metric 5 hostname(config-route-map)# set metric-type type-1 hostname(config-router)# redistribute ospf 1 route-map 1-to-2</td>
<td></td>
</tr>
<tr>
<td>redistribute eigrp as-num [metric metric-value] [metric-type {type-1</td>
<td>type-2}] [tag tag_value] [subnets] [route-map map_name]</td>
</tr>
<tr>
<td><strong>Example:</strong> hostname(config)# redistribute eigrp 2 hostname(config-route-map)# match metric 1 hostname(config-route-map)# set metric 5 hostname(config-route-map)# set metric-type type-1 hostname(config-router)# redistribute ospf 1 route-map 1-to-2</td>
<td></td>
</tr>
</tbody>
</table>
To configure the software advertisement on one summary route for all redistributed routes included for a network address and mask, perform the following steps:

**Detailed Steps**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><strong>router ospf process_id</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname(config)# router ospf 1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 2</strong></td>
<td><strong>summary-address ip_address mask [not-advertise] [tag tag]</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname(config)# router ospf 1 hostname(config-router)# summary-address 10.1.0.0 255.255.0.0</td>
</tr>
</tbody>
</table>

**Configuring Route Summarization Between OSPF Areas**

Route summarization is the consolidation of advertised addresses. This feature causes a single summary route to be advertised to other areas by an area boundary router. In OSPF, an area boundary router advertises networks in one area into another area. If the network numbers in an area are assigned in a way so that they are contiguous, you can configure the area boundary router to advertise a summary route that includes all the individual networks within the area that fall into the specified range.

To define an address range for route summarization, perform the following steps:

**Detailed Steps**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><strong>router ospf process_id</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname(config)# router ospf 1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 2</strong></td>
<td>**area area-id range ip-address mask [advertise</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname(config)# router ospf 1 hostname(config-router)# area 17 range 12.1.0.0 255.255.0.0</td>
</tr>
</tbody>
</table>
Customizing OSPF

You can change some interface-specific OSPF parameters, if necessary.

Prerequisites

You are not required to change any of these parameters, but the following interface parameters must be consistent across all routers in an attached network: `ospf hello-interval`, `ospf dead-interval`, and `ospf authentication-key`. If you configure any of these parameters, be sure that the configurations for all routers on your network have compatible values.

To configure OSPF interface parameters, perform the following steps:

## Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>router ospf process_id</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# router ospf 2</td>
</tr>
<tr>
<td></td>
<td>Creates an OSPF routing process and enters router configuration mode for the OSPF process that you want to redistribute. The <code>process_id</code> argument is an internally used identifier for this routing process and can be any positive integer. This ID does not have to match the ID on any other device; it is for internal use only. You can use a maximum of two processes.</td>
</tr>
</tbody>
</table>

| **Step 2** | network ip_address mask area area_id |
| Example: | hostname(config)# router ospf 2 hostname(config-router)# network 10.0.0.0 255.0.0.0 area 0 |
| | Defines the IP addresses on which OSPF runs and the area ID for that interface. |

| **Step 3** | hostname(config)# interface interface_name |
| Example: | hostname(config)# interface my_interface |
| | Allows you to enter interface configuration mode. |

<p>| <strong>Step 4</strong> | Do one of the following to configure optional OSPF interface parameters: |
| | ospf authentication [message-digest | null] |
| Example: | hostname(config-interface)# ospf authentication message-digest |
| | Specifies the authentication type for an interface. |</p>
<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ospf authentication-key</strong> <code>key</code></td>
<td>Allows you to assign a password to be used by neighboring OSPF routers on a network segment that is using the OSPF simple password authentication. The <code>key</code> argument can be any continuous string of characters up to 8 bytes in length. The password created by this command is used as a key that is inserted directly into the OSPF header when the ASASM software originates routing protocol packets. A separate password can be assigned to each network on a per-interface basis. All neighboring routers on the same network must have the same password to be able to exchange OSPF information.</td>
</tr>
<tr>
<td><strong>Example:</strong>&lt;br&gt;<code>hostname(config-interface)# ospf authentication-key cisco</code></td>
<td></td>
</tr>
<tr>
<td><strong>ospf cost</strong> <code>cost</code></td>
<td>Allows you to explicitly specify the cost of sending a packet on an OSPF interface. The <code>cost</code> is an integer from 1 to 65535. In this example, the cost is set to 20.</td>
</tr>
<tr>
<td><strong>Example:</strong>&lt;br&gt;<code>hostname(config-interface)# ospf cost 20</code></td>
<td></td>
</tr>
<tr>
<td><strong>ospf dead-interval</strong> <code>seconds</code></td>
<td>Allows you to set the number of seconds that a device must wait before it declares a neighbor OSPF router down because it has not received a hello packet. The value must be the same for all nodes on the network. In this example, the dead interval is set to 40.</td>
</tr>
<tr>
<td><strong>Example:</strong>&lt;br&gt;<code>hostname(config-interface)# ospf dead-interval 40</code></td>
<td></td>
</tr>
<tr>
<td><strong>ospf hello-interval</strong> <code>seconds</code></td>
<td>Allows you to specify the length of time between the hello packets that the ASASM sends on an OSPF interface. The value must be the same for all nodes on the network. In this example, the hello interval is set to 10.</td>
</tr>
<tr>
<td><strong>Example:</strong>&lt;br&gt;<code>hostname(config-interface)# ospf hello-interval 10</code></td>
<td></td>
</tr>
</tbody>
</table>
| **ospf message-digest-key** `key_id` `md5` `key` | Enables OSPF MD5 authentication. The following argument values can be set:  
- `key_id`—An identifier in the range from 1 to 255.  
- `key`—An alphanumeric password of up to 16 bytes. Usually, one key per interface is used to generate authentication information when sending packets and to authenticate incoming packets. The same key identifier on the neighbor router must have the same key value. We recommend that you not keep more than one key per interface. Every time you add a new key, you should remove the old key to prevent the local system from continuing to communicate with a hostile system that knows the old key. Removing the old key also reduces overhead during rollover. |
| **Example:**<br>`hostname(config-interface)# ospf message-digest-key 1 md5 cisco` | |
| **ospf priority** `number_value` | Allows you to set the priority to help determine the OSPF designated router for a network. The `number_value` argument ranges from 0 to 255. In this example, the priority number value is set to 20. |
| **Example:**<br>`hostname(config-interface)# ospf priority 20` | |
Customizing OSPF

Configuring OSPF Area Parameters

You can configure several OSPF area parameters. These area parameters (shown in the following task list) include setting authentication, defining stub areas, and assigning specific costs to the default summary route. Authentication provides password-based protection against unauthorized access to an area.

Stub areas are areas into which information on external routes is not sent. Instead, there is a default external route generated by the ABR into the stub area for destinations outside the autonomous system. To take advantage of the OSPF stub area support, default routing must be used in the stub area. To further reduce the number of LSAs sent into a stub area, you can use the `no-summary` keyword of the `area` command on the ABR to prevent it from sending a summary link advertisement (LSA Type 3) into the stub area.

To specify area parameters for your network, perform the following steps:

### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong>&lt;br&gt;<code>router ospf process_id</code>&lt;br&gt;例：<code>hostname(config)# router ospf 2</code></td>
<td>创建一个OSPF路由进程并进入路由配置模式，用于你的OSPF进程，你希望重新分配。<code>process_id</code>参数是一个内部使用的标识符，该标识符可以是任何正数。此ID不需要与其他设备匹配；它仅用于内部使用。你可以使用最多两个进程。</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>配置可选的OSPF区域参数。</td>
</tr>
</tbody>
</table>
Customizing OSPF

Configuring OSPF NSSA

The OSPF implementation of an NSSA is similar to an OSPF stub area. NSSA does not flood Type 5 external LSAs from the core into the area, but it can import autonomous system external routes in a limited way within the area.

NSSA imports Type 7 autonomous system external routes within an NSSA area by redistribution. These Type 7 LSAs are translated into Type 5 LSAs by NSSA ABRs, which are flooded throughout the whole routing domain. Summarization and filtering are supported during the translation.

You can simplify administration if you are an ISP or a network administrator that must connect a central site using OSPF to a remote site that is using a different routing protocol using NSSA.

Before the implementation of NSSA, the connection between the corporate site border router and the remote router could not be run as an OSPF stub area because routes for the remote site could not be redistributed into the stub area, and two routing protocols needed to be maintained. A simple protocol such as RIP was usually run and handled the redistribution. With NSSA, you can extend OSPF to cover the remote connection by defining the area between the corporate router and the remote router as an NSSA.

Before you use this feature, consider these guidelines:

- You can set a Type 7 default route that can be used to reach external destinations. When configured, the router generates a Type 7 default into the NSSA or the NSSA area boundary router.
- Every router within the same area must agree that the area is NSSA; otherwise, the routers will not be able to communicate.

To specify area parameters for your network to configure OSPF NSSA, perform the following steps:

### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>area area-id</td>
<td>Enables authentication for an OSPF area.</td>
</tr>
<tr>
<td>area area-id</td>
<td>Enables authentication for an OSPF area.</td>
</tr>
<tr>
<td>area area-id</td>
<td>Enables MD5 authentication for an OSPF area.</td>
</tr>
<tr>
<td>area area-id</td>
<td>Enables MD5 authentication for an OSPF area.</td>
</tr>
</tbody>
</table>

**Step 1**

```
router ospf process_id
```

Creates an OSPF routing process and enters router configuration mode for the OSPF process that you want to redistribute.

The `process_id` argument is an internally used identifier for this routing process. It can be any positive integer. This ID does not have to match the ID on any other device; it is for internal use only. You can use a maximum of two processes.

**Step 2**

Do one of the following to configure optional OSPF NSSA parameters:
## Customizing OSPF

**Note**

OSPF does not support summary-address 0.0.0.0 0.0.0.0.

### Defining Static OSPF Neighbors

You need to define static OSPF neighbors to advertise OSPF routes over a point-to-point, non-broadcast network. This feature lets you broadcast OSPF advertisements across an existing VPN connection without having to encapsulate the advertisements in a GRE tunnel.

Before you begin, you must create a static route to the OSPF neighbor. See Chapter 20, “Configuring Static and Default Routes,” for more information about creating static routes.

To define a static OSPF neighbor, perform the following steps:

#### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><strong>Step 2</strong></td>
</tr>
<tr>
<td><code>router ospf process_id</code></td>
<td><code>neighbor addr [interface if_name]</code></td>
</tr>
<tr>
<td>Creates an OSPF routing process and enters router configuration mode for this OSPF process.</td>
<td>Defines the OSPF neighborhood.</td>
</tr>
<tr>
<td>The <code>process_id</code> argument is an internally used identifier for this routing process and can be any positive integer. This ID does not have to match the ID on any other device; it is for internal use only. You can use a maximum of two processes.</td>
<td>The <code>addr</code> argument is the IP address of the OSPF neighbor. The <code>if_name</code> argument is the interface used to communicate with the neighbor. If the OSPF neighbor is not on the same network as any of the directly connected interfaces, you must specify the interface.</td>
</tr>
<tr>
<td><code>hostname(config)# router ospf 2</code></td>
<td><code>hostname(config-router)# neighbor 255.255.0.0 [interface my_interface]</code></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>area area-id nssa [no-redistribution] [default-information-originate]</code></td>
<td>Defines an NSSA area.</td>
</tr>
<tr>
<td>Example: <code>hostname(config-router)# area 0 nssa</code></td>
<td>Sets the summary address and helps reduce the size of the routing table. Using this command for OSPF causes an OSPF ASBR to advertise one external route as an aggregate for all redistributed routes that are covered by the address.</td>
</tr>
<tr>
<td><code>summary-address ip_address mask [not-advertise] [tag tag]</code></td>
<td>In this example, the summary address 10.1.0.0 includes addresses 10.1.1.0, 10.1.2.0, 10.1.3.0, and so on. Only the 10.1.0.0 address is advertised in an external link-state advertisement.</td>
</tr>
<tr>
<td>Example: <code>hostname(config)# router ospf 1</code> <code>hostname(config-router)# summary-address 10.1.0.0 255.255.0.0</code></td>
<td></td>
</tr>
</tbody>
</table>
Configuring Route Calculation Timers

You can configure the delay time between when OSPF receives a topology change and when it starts an SPF calculation. You also can configure the hold time between two consecutive SPF calculations.

To configure route calculation timers, perform the following steps:

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>router ospf process_id</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# router ospf 2</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>timers spf spf-delay spf-holdtime</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config-router)# timers spf 10 120</td>
</tr>
</tbody>
</table>

Logging Neighbors Going Up or Down

By default, a syslog message is generated when an OSPF neighbor goes up or down.

Configure log-adj-changes router configuration command if you want to know about OSPF neighbors going up or down without turning on the debug ospf adjacency command. The log-adj-changes router configuration command provides a higher level view of the peer relationship with less output. Configure the log-adj-changes detail command if you want to see messages for each state change.
To log neighbors going up or down, perform the following steps:

### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td><code>router ospf process_id</code></td>
<td>Creates an OSPF routing process and enters router configuration mode for this OSPF process. The <code>process_id</code> argument is an internally used identifier for this routing process and can be any positive integer. This ID does not have to match the ID on any other device; it is for internal use only. You can use a maximum of two processes.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td><code>hostname(config)# router ospf 2</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td><code>log-adj-changes [detail]</code></td>
<td>Configures logging for neighbors going up or down.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td><code>hostname(config-router)# log-adj-changes [detail]</code></td>
<td></td>
</tr>
</tbody>
</table>

### Restarting the OSPF Process

To remove the entire OSPF configuration that you have enabled, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`clear ospf pid {process</td>
<td>redistribution</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td><code>hostname(config)# clear ospf</code></td>
<td></td>
</tr>
</tbody>
</table>

### Configuration Example for OSPF

The following example shows how to enable and configure OSPF with various optional processes:

**Step 1**  
To enable OSPF, enter the following commands:

```plaintext
hostname(config)# router ospf 2  
hostname(config-router)# network 10.0.0.0 255.0.0.0 area 0
```

**Step 2**  
(Optional) To redistribute routes from one OSPF process to another OSPF process, enter the following commands:

```plaintext
hostname(config)# route-map 1-to-2 permit  
hostname(config-route-map)# match metric 1  
hostname(config-route-map)# set metric 5  
hostname(config-route-map)# set metric-type type-1  
hostname(config-route-map)# router ospf 2  
hostname(config-router)# redistribute ospf 1 route-map 1-to-2
```
Chapter 22  Configuring OSPF

Configuration Example for OSPF

Step 3  (Optional) To configure OSPF interface parameters, enter the following commands:

```
hostname(config)# router ospf 2
hostname(config-router)# network 10.0.0.0 255.0.0.0 area 0
hostname(config-router)# interface inside
hostname(config-interface)# ospf cost 20
hostname(config-interface)# ospf retransmit-interval 15
hostname(config-interface)# ospf transmit-delay 10
hostname(config-interface)# ospf priority 20
hostname(config-interface)# ospf hello-interval 10
hostname(config-interface)# ospf dead-interval 40
hostname(config-interface)# ospf authentication-key cisco
hostname(config-interface)# ospf message-digest-key 1 md5 cisco
hostname(config-interface)# ospf authentication message-digest
```

Step 4  (Optional) To configure OSPF area parameters, enter the following commands:

```
hostname(config)# router ospf 2
hostname(config-router)# area 0 authentication
hostname(config-router)# area 0 authentication message-digest
hostname(config-router)# area 17 stub
hostname(config-router)# area 17 default-cost 20
```

Step 5  (Optional) To configure the route calculation timers and show the log neighbor up and down messages, enter the following commands:

```
hostname(config-router)# timers spf 10 120
hostname(config-router)# log-adj-changes [detail]
```

Step 6  To restart the OSPF process, enter the following commands:

```
hostname(config)# clear ospf pid {
  process | redistribution | counters
  [neighbor [neighbor-interface] [neighbor-id]]
```

Step 7  (Optional) To show current OSPF configuration settings, enter the `show ospf` command.
The following is sample output from the `show ospf` command:

```
hostname(config)# show ospf
Routing Process "ospf 2" with ID 10.1.89.2 and Domain ID 0.0.0.2
Supports only single TOS(TOS0) routes
Supports opaque LSA
SPF schedule delay 5 secs, Hold time between two SPF 10 secs
Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs
Number of external LSA 5. Checksum Sum 0x 26da6
Number of opaque AS LSA 0. Checksum Sum 0x 0
Number of DCbitless external and opaque AS LSA 0
Number of DoNotAge external and opaque AS LSA 0
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
External flood list length 0
Area BACKBONE(0):
  Number of interfaces in this area is 1
  Area has no authentication
  SPF algorithm executed 2 times
  Area ranges are
  Number of LSA 5. Checksum Sum 0x 209a3
  Number of opaque link LSA 0. Checksum Sum 0x 0
  Number of DCbitless LSA 0
  Number of indication LSA 0
  Number of DoNotAge LSA 0
  Flood list length 0
```
Monitoring OSPF

You can display specific statistics such as the contents of IP routing tables, caches, and databases. You can also use the information provided to determine resource utilization and solve network problems. You can also display information about node reachability and discover the routing path that your device packets are taking through the network.

To monitor or display various OSPF routing statistics, enter one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show ospf [process-id [area-id]]</code></td>
<td>Displays general information about OSPF routing processes.</td>
</tr>
<tr>
<td><code>show ospf border-routers</code></td>
<td>Displays the internal OSPF routing table entries to the ABR and ASBR.</td>
</tr>
<tr>
<td><code>show ospf [process-id [area-id]] database</code></td>
<td>Displays lists of information related to the OSPF database for a specific router.</td>
</tr>
<tr>
<td><code>show ospf flood-list if-name</code></td>
<td>Displays a list of LSAs waiting to be flooded over an interface (to observe OSPF packet pacing). OSPF update packets are automatically paced so they are not sent less than 33 milliseconds apart. Without pacing, some update packets could get lost in situations where the link is slow, a neighbor could not receive the updates quickly enough, or the router could run out of buffer space. For example, without pacing, packets might be dropped if either of the following topologies exist:</td>
</tr>
<tr>
<td></td>
<td>- A fast router is connected to a slower router over a point-to-point link.</td>
</tr>
<tr>
<td></td>
<td>- During flooding, several neighbors send updates to a single router at the same time.</td>
</tr>
<tr>
<td></td>
<td>Pacing is also used between resends to increase efficiency and minimize lost retransmissions. You also can display the LSAs waiting to be sent out of an interface. Pacing enables OSPF update and retransmission packets to be sent more efficiently. There are no configuration tasks for this feature; it occurs automatically.</td>
</tr>
<tr>
<td><code>show ospf interface [if_name]</code></td>
<td>Displays OSPF-related interface information.</td>
</tr>
<tr>
<td><code>show ospf neighbor [interface-name] [neighbor-id] [detail]</code></td>
<td>Displays OSPF neighbor information on a per-interface basis.</td>
</tr>
<tr>
<td><code>show ospf request-list neighbor if_name</code></td>
<td>Displays a list of all LSAs requested by a router.</td>
</tr>
<tr>
<td><code>show ospf retransmission-list neighbor if_name</code></td>
<td>Displays a list of all LSAs waiting to be resent.</td>
</tr>
</tbody>
</table>
Feature History for OSPF

Table 22-1 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSPF support</td>
<td>7.0(1)</td>
<td>Support was added for route data, authentication, and redistribution and monitoring of routing information using the Open Shortest Path First (OSPF) routing protocol. We introduced the <code>route ospf</code> command.</td>
</tr>
</tbody>
</table>
Configuring RIP

This chapter describes how to configure the ASASM to route data, perform authentication, and redistribute routing information using the Routing Information Protocol (RIP).

This chapter includes the following sections:

- Information About RIP, page 23-1
- Licensing Requirements for RIP, page 23-3
- Guidelines and Limitations, page 23-3
- Configuring RIP, page 23-4
- Customizing RIP, page 23-4
- Monitoring RIP, page 23-11
- Configuration Example for RIP, page 23-11
- Feature History for RIP, page 23-11

Information About RIP

This section includes the following topics:

- Routing Update Process, page 23-2
- RIP Routing Metric, page 23-2
- RIP Stability Features, page 23-2
- RIP Timers, page 23-2

The Routing Information Protocol, or RIP, as it is more commonly called, is one of the most enduring of all routing protocols. RIP has four basic components: routing update process, RIP routing metrics, routing stability, and routing timers. Devices that support RIP send routing-update messages at regular intervals and when the network topology changes. These RIP packets include information about the networks that the devices can reach, as well as the number of routers or gateways that a packet must travel through to reach the destination address. RIP generates more traffic than OSPF, but is easier to configure.

RIP is a distance-vector routing protocol that uses hop count as the metric for path selection. When RIP is enabled on an interface, the interface exchanges RIP broadcasts with neighboring devices to dynamically learn about and advertise routes.
The ASASM supports both RIP Version 1 and RIP Version 2. RIP Version 1 does not send the subnet mask with the routing update. RIP Version 2 sends the subnet mask with the routing update and supports variable-length subnet masks. Additionally, RIP Version 2 supports neighbor authentication when routing updates are exchanged. This authentication ensures that the ASASM receives reliable routing information from a trusted source.

RIP has advantages over static routes because the initial configuration is simple, and you do not need to update the configuration when the topology changes. The disadvantage to RIP is that there is more network and processing overhead than in static routing.

**Routing Update Process**

RIP sends routing-update messages at regular intervals and when the network topology changes. When a router receives a routing update that includes changes to an entry, it updates its routing table to reflect the new route. The metric value for the path is increased by 1, and the sender is indicated as the next hop. RIP routers maintain only the best route (the route with the lowest metric value) to a destination. After updating its routing table, the router immediately begins transmitting routing updates to inform other network routers of the change. These updates are sent independently of the regularly scheduled updates that RIP routers send.

**RIP Routing Metric**

RIP uses a single routing metric (hop count) to measure the distance between the source and a destination network. Each hop in a path from source to destination is assigned a hop count value, which is typically 1. When a router receives a routing update that contains a new or changed destination network entry, the router adds 1 to the metric value indicated in the update and enters the network in the routing table. The IP address of the sender is used as the next hop.

**RIP Stability Features**

RIP prevents routing loops from continuing indefinitely by implementing a limit on the number of hops allowed in a path from the source to a destination. The maximum number of hops in a path is 15. If a router receives a routing update that contains a new or changed entry, and if increasing the metric value by 1 causes the metric to be infinity (that is, 16), the network destination is considered unreachable. The downside of this stability feature is that it limits the maximum diameter of a RIP network to less than 16 hops.

RIP includes a number of other stability features that are common to many routing protocols. These features are designed to provide stability despite potentially rapid changes in network topology. For example, RIP implements the split horizon and hold-down mechanisms to prevent incorrect routing information from being propagated.

**RIP Timers**

RIP uses numerous timers to regulate its performance. These include a routing-update timer, a route-timeout timer, and a route-flush timer. The routing-update timer clocks the interval between periodic routing updates. Generally, it is set to 30 seconds, with a small random amount of time added whenever the timer is reset. This is done to help prevent congestion, which could result from all routers
simultaneously attempting to update their neighbors. Each routing table entry has a route-timeout timer associated with it. When the route-timeout timer expires, the route is marked invalid but is retained in the table until the route-flush timer expires.

**Licensing Requirements for RIP**

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License</td>
</tr>
</tbody>
</table>

**Guidelines and Limitations**

This section includes the guidelines and limitations for this feature.

**Context Mode Guidelines**
Supported in single context mode only.

**Firewall Mode Guidelines**
Supported in routed and transparent firewall mode.

**IPv6 Guidelines**
Does not support IPv6.

**Additional Guidelines**
The following information applies to RIP Version 2 only:

- If using neighbor authentication, the authentication key and key ID must be the same on all neighbor devices that provide RIP Version 2 updates to the interface.
- With RIP Version 2, the ASASM transmits and receives default route updates using the multicast address 224.0.0.9. In passive mode, it receives route updates at that address.
- When RIP Version 2 is configured on an interface, the multicast address 224.0.0.9 is registered on that interface. When a RIP Version 2 configuration is removed from an interface, that multicast address is unregistered.

**Limitations**
- The ASASM cannot pass RIP updates between interfaces.
- RIP Version 1 does not support variable-length subnet masks.
- RIP has a maximum hop count of 15. A route with a hop count greater than 15 is considered unreachable.
- RIP convergence is relatively slow compared to other routing protocols.
- You can only enable a single RIP process on the ASASM.
Chapter 23  Configuring RIP

Configuring RIP

This section describes how to enable and restart the RIP process on the ASASM.

After you have enabled RIP, see the “Customizing RIP” section on page 23-4 to learn how to customize the RIP process on the ASASM.

Note

If you want to redistribute a route by defining which of the routes from the specified routing protocol are allowed to be redistributed into the target routing process, you must first generate a default route. For information, see the “Configuring a Default Static Route” section on page 20-4 and then define a route map. For information, see the “Defining a Route Map” section on page 21-4.

Enabling RIP

You can only enable one RIP routing process on the ASASM. After you enable the RIP routing process, you must define the interfaces that will participate in that routing process using the `network` command. By default, the ASASM sends RIP Version 1 updates and accepts RIP Version 1 and Version 2 updates.

To enable the RIP routing process, enter the following command:

```
router rip
```

Example:

```
hostname(config)# router rip
```

Starts the RIP routing process and places you in router configuration mode.

Use the `no router rip` command to remove the entire RIP configuration that you have enabled. After the configuration is cleared, you must reconfigure RIP using the `router rip` command.

Customizing RIP

This section describes how to configure RIP and includes the following topics:

- Configuring the RIP Version, page 23-5
- Configuring Interfaces for RIP, page 23-6
- Configuring the RIP Send and Receive Version on an Interface, page 23-6
- Configuring Route Summarization, page 23-7
- Filtering Networks in RIP, page 23-8
- Redistributing Routes into the RIP Routing Process, page 23-8
- Enabling RIP Authentication, page 23-9
- Restarting the RIP Process, page 23-10
# Configuring the RIP Version

To specify the version of RIP used by the ASASM, perform the following steps:

## Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td><code>router rip</code></td>
<td>Starts the RIP routing process and places you in router configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# router rip</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td><code>network network_address</code></td>
<td>Specifies the interfaces that will participate in the RIP routing process.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# router rip</td>
<td></td>
</tr>
<tr>
<td>hostname(config-router)# network 10.0.0.0</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 3</strong></td>
<td></td>
</tr>
<tr>
<td>`version [1</td>
<td>2]`</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config-router):# version [1]</td>
<td>In this example, Version 1 is entered.</td>
</tr>
</tbody>
</table>
Configuring Interfaces for RIP

If you have an interface that you do not want to have participate in RIP routing, but that is attached to a network that you want advertised, you can configure the network (using the network command) that includes the network to which the interface is attached, and configure the passive interfaces (using the passive-interface command) to prevent that interface from using RIP. Additionally, you can specify the version of RIP that is used by the ASASM for updates.

To configure interfaces for RIP, perform the following steps:

**Detailed Steps**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 router rip</td>
<td>Starts the RIP routing process and places you in router configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# router rip</td>
<td></td>
</tr>
<tr>
<td>Step 2 network network_address</td>
<td>Specifies the interfaces that will participate in the RIP routing process. If an interface belongs to a network defined by this command, the interface will participate in the RIP routing process. If an interface does not belong to a network defined by this command, it will not send or receive RIP updates.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# router rip</td>
<td></td>
</tr>
<tr>
<td>hostname(config-router)# network 10.0.0.0</td>
<td></td>
</tr>
<tr>
<td>Step 3 passive-interface [default</td>
<td>If an interface belongs to a network defined by this command, the interface will participate in the RIP routing process. If an interface does not belong to a network defined by this command, it will not send or receive RIP updates.</td>
</tr>
<tr>
<td>[if_name]</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config-router)# passive-interface [default]</td>
<td>Specifies an interface to operate in passive mode. Using the default keyword causes all interfaces to operate in passive mode. Specifying an interface name sets only that interface to passive mode. In passive mode, RIP routing updates are accepted by, but not sent out of, the specified interface. You can enter this command for each interface that you want to set to passive mode.</td>
</tr>
</tbody>
</table>

Configuring the RIP Send and Receive Version on an Interface

You can override the globally-set version of RIP that the ASASM uses to send and receive RIP updates on a per-interface basis.

To configure the RIP version for sending and receiving updates, perform the following steps:

**Detailed Steps**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 interface phy_if</td>
<td>Enters interface configuration mode for the interface that you are configuring.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# interface phy_if</td>
<td></td>
</tr>
<tr>
<td>Step 2</td>
<td>Do one of the following to send or receive RIP updates on a per-interface basis.</td>
</tr>
</tbody>
</table>


Chapter 23      Configuring RIP

Customizing RIP

Configuring Route Summarization

Note

The RIP routing process summarizes on network number boundaries, which can cause routing problems if you have noncontiguous networks.

For example, if you have a router with the networks 192.168.1.0, 192.168.2.0, and 192.168.3.0 connected to it, and those networks all participate in RIP, the RIP routing process creates the summary address 192.168.0.0 for those routes. If an additional router is added to the network with the networks 192.168.10.0 and 192.168.11.0, and those networks participate in RIP, they will also be summarized as 192.168.0.0. To prevent the possibility of traffic being routed to the wrong location, you should disable automatic route summarization on the routers that are creating conflicting summary addresses.

Because RIP Version 1 always uses automatic route summarization, and RIP Version 2 always uses automatic route summarization by default, when configuring automatic route summarization, you only need to disable it.

To disable automatic route summarization, enter the following command:

**Detailed Steps**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong>&lt;br&gt;router rip</td>
<td>Enables the RIP routing process and places you in router configuration mode.</td>
</tr>
<tr>
<td><strong>Step 2</strong>&lt;br&gt;no auto-summarize</td>
<td>Disables automatic route summarization.</td>
</tr>
</tbody>
</table>

For example:

```
hostname(config)# router rip
```

```
hostname(config-router):# no auto-summarize
```
Filtering Networks in RIP

To filter the networks received in updates, perform the following steps:

Note: Before you begin, you must create a standard access list that permits the networks that you want the RIP process to allow in the routing table and denies the networks that you want the RIP process to discard.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><strong>router rip</strong>&lt;br&gt;&lt;br&gt;Example: hostname(config)# router rip</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><strong>distribute-list acl in [interface if_name]</strong>&lt;br&gt;distribute-list acl out [connected</td>
</tr>
</tbody>
</table>

| Example: | hostname(config-router)# distribute-list acl2 in [interface interface1]<br>hostname(config-router)# distribute-list acl3 out [connected] | |

Redistributing Routes into the RIP Routing Process

You can redistribute routes from the OSPF, EIGRP, static, and connected routing processes into the RIP routing process.

Note: Before you begin this procedure, you must create a route map to further define which routes from the specified routing protocol are redistributed into the RIP routing process. See Chapter 21, “Defining a Route Map,” for more information about creating a route map.
To redistribute a route into the RIP routing process, enter one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>redistribute connected [metric metric-value</td>
<td>transparent] [route-map route-map-name]</td>
</tr>
<tr>
<td>redistribute static [metric {metric_value</td>
<td>transparent}] [route-map map_name]</td>
</tr>
<tr>
<td>redistribute ospf pid [match {internal</td>
<td>external [1</td>
</tr>
<tr>
<td>redistribute eigrp as-num [metric {metric_value</td>
<td>transparent}] [route-map map_name]</td>
</tr>
</tbody>
</table>

### Enabling RIP Authentication

**Note**  The ASASM supports RIP message authentication for RIP Version 2 messages.
RIP route authentication provides MD5 authentication of routing updates from the RIP routing protocol. The MD5 keyed digest in each RIP packet prevents the introduction of unauthorized or false routing messages from unapproved sources.

RIP route authentication is configured on a per-interface basis. All RIP neighbors on interfaces configured for RIP message authentication must be configured with the same authentication mode and key for adjacencies to be established.

**Note**

Before you can enable RIP route authentication, you must enable RIP.

To enable RIP authentication on an interface, perform the following steps:

### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> router rip <strong>as-num</strong></td>
<td>Creates the RIP routing process and enters router configuration mode for this RIP process. The <code>as-num</code> argument is the autonomous system number of the RIP routing process.</td>
</tr>
<tr>
<td><strong>Example:</strong> hostname(config)# router rip 2</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> interface <strong>phy_if</strong></td>
<td>Enters interface configuration mode for the interface on which you are configuring RIP message authentication.</td>
</tr>
<tr>
<td><strong>Example:</strong> hostname(config)# interface phy_if</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> rip authentication mode (<strong>text</strong></td>
<td>md5)</td>
</tr>
<tr>
<td><strong>Example:</strong> hostname(config-if)# rip authentication mode md5</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> rip authentication key <strong>key</strong> <strong>key-id</strong> <strong>key-id</strong></td>
<td>Configures the authentication key used by the MD5 algorithm. The <code>key</code> argument can include up to 16 characters. The <code>key-id</code> argument is a number from 0 to 255.</td>
</tr>
<tr>
<td><strong>Example:</strong> hostname(config-if)# rip authentication key cisco key-id 200</td>
<td></td>
</tr>
</tbody>
</table>

### Restarting the RIP Process

To remove the entire RIP configuration, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>clear rip <strong>pid</strong> (<strong>process</strong></td>
<td>redistribution</td>
</tr>
<tr>
<td><strong>Example:</strong> hostname(config)# clear rip</td>
<td></td>
</tr>
</tbody>
</table>
Monitoring RIP

We recommend that you only use the `debug` commands to troubleshoot specific problems or during troubleshooting sessions with the Cisco TAC.

Debugging output is assigned high priority in the CPU process and can render the ASASM unusable. It is best to use `debug` commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased `debug` command processing overhead will affect performance. For examples and descriptions of the command output, see the command reference.

To monitor or debug various RIP routing statistics, enter one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Monitoring RIP Routing</strong></td>
<td></td>
</tr>
<tr>
<td><code>show rip database</code></td>
<td>Display the contents of the RIP routing database.</td>
</tr>
<tr>
<td><code>show running-config router rip</code></td>
<td>Displays the RIP commands.</td>
</tr>
<tr>
<td><strong>Debugging RIP</strong></td>
<td></td>
</tr>
<tr>
<td><code>debug rip events</code></td>
<td>Displays RIP processing events.</td>
</tr>
<tr>
<td><code>debug rip database</code></td>
<td>Displays RIP database events.</td>
</tr>
</tbody>
</table>

Configuration Example for RIP

The following example shows how to enable and configure RIP with various optional processes:

```
hostname(config)# router rip 2
hostname(config-router)# default-information originate
hostname(config-router)# version [1]
hostname(config-router)# network 225.25.25.225
hostname(config-router)# passive-interface [default]
hostname(config-router)# redistribute connected [metric bandwidth delay reliability loading mtu] [route-map map_name]
```

Feature History for RIP

Table 23-1 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>RIP support</td>
<td>7.0(1)</td>
<td>Support was added for routing data, performing authentication, and redistributing and monitoring routing information using the Routing Information Protocol (RIP). We introduced the <code>route rip</code> command.</td>
</tr>
</tbody>
</table>
Configuring Multicast Routing

This chapter describes how to configure the ASASM to use the multicast routing protocol and includes the following sections:

- Information About Multicast Routing, page 24-1
- Licensing Requirements for Multicast Routing, page 24-2
- Guidelines and Limitations, page 24-3
- Enabling Multicast Routing, page 24-3
- Customizing Multicast Routing, page 24-4
- Configuration Example for Multicast Routing, page 24-14
- Additional References, page 24-15
- Feature History for Multicast Routing, page 24-15

Information About Multicast Routing

Multicast routing is a bandwidth-conserving technology that reduces traffic by simultaneously delivering a single stream of information to thousands of corporate recipients and homes. Applications that take advantage of multicast routing include videoconferencing, corporate communications, distance learning, and distribution of software, stock quotes, and news.

Multicast routing protocols delivers source traffic to multiple receivers without adding any additional burden on the source or the receivers while using the least network bandwidth of any competing technology. Multicast packets are replicated in the network by Cisco routers enabled with Protocol Independent Multicast (PIM) and other supporting multicast protocols resulting in the most efficient delivery of data to multiple receivers possible.

The ASASM supports both stub multicast routing and PIM multicast routing. However, you cannot configure both concurrently on a single ASASM.

The UDP and non-UDP transports are both supported for multicast routing. However, the non-UDP transport has no FastPath optimization.

This section includes the following topics:

- Stub Multicast Routing, page 24-2
- PIM Multicast Routing, page 24-2
Stub Multicast Routing

Stub multicast routing provides dynamic host registration and facilitates multicast routing. When configured for stub multicast routing, the ASASM acts as an IGMP proxy agent. Instead of fully participating in multicast routing, the ASASM forwards IGMP messages to an upstream multicast router, which sets up delivery of the multicast data. When configured for stub multicast routing, the ASASM cannot be configured for PIM.

The ASASM supports both PIM-SM and bidirectional PIM. PIM-SM is a multicast routing protocol that uses the underlying unicast routing information base or a separate multicast-capable routing information base. It builds unidirectional shared trees rooted at a single Rendezvous Point per multicast group and optionally creates shortest-path trees per multicast source.

PIM Multicast Routing

Bi-directional PIM is a variant of PIM-SM that builds bi-directional shared trees connecting multicast sources and receivers. Bi-directional trees are built using a DF election process operating on each link of the multicast topology. With the assistance of the DF, multicast data is forwarded from sources to the Rendezvous Point, and therefore along the shared tree to receivers, without requiring source-specific state. The DF election takes place during Rendezvous Point discovery and provides a default route to the Rendezvous Point.

Note

If the ASASM is the PIM RP, use the untranslated outside address of the ASASM as the RP address.

Multicast Group Concept

Multicast is based on the concept of a group. An arbitrary group of receivers expresses an interest in receiving a particular data stream. This group does not have any physical or geographical boundaries—the hosts can be located anywhere on the Internet. Hosts that are interested in receiving data flowing to a particular group must join the group using IGMP. Hosts must be a member of the group to receive the data stream.

Multicast Addresses

Multicast addresses specify an arbitrary group of IP hosts that have joined the group and want to receive traffic sent to this group.

Licensing Requirements for Multicast Routing

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>
Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

**Context Mode Guidelines**
Supported in single context mode. In multiple context mode, unshared interfaces and shared interfaces are not supported.

**Firewall Mode Guidelines**
Supported only in routed firewall mode. Transparent firewall mode is not supported.

**IPv6 Guidelines**
Does not support IPv6.

Enabling Multicast Routing

Enabling multicast routing lets you enable multicast routing on the ASASM. Enabling multicast routing enables IGMP and PIM on all interfaces by default. IGMP is used to learn whether members of a group are present on directly attached subnets. Hosts join multicast groups by sending IGMP report messages. PIM is used to maintain forwarding tables to forward multicast datagrams.

**Note**
Only the UDP transport layer is supported for multicast routing.

To enable multicast routing, enter the following command:

```
hostname(config)# multicast-routing
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>multicast-routing</td>
<td>Enables multicast routing.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Example:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>hostname(config)# multicast-routing</td>
<td></td>
</tr>
</tbody>
</table>

Table 24-1 lists the maximum number of entries for specific multicast tables based on the amount of RAM on the ASASM. Once these limits are reached, any new entries are discarded.

**Table 24-1**  Entry Limits for Multicast Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>16 MB</th>
<th>128 MB</th>
<th>128+ MB</th>
</tr>
</thead>
<tbody>
<tr>
<td>MFIB</td>
<td>1000</td>
<td>3000</td>
<td>5000</td>
</tr>
<tr>
<td>IGMP Groups</td>
<td>1000</td>
<td>3000</td>
<td>5000</td>
</tr>
<tr>
<td>PIM Routes</td>
<td>3000</td>
<td>7000</td>
<td>12000</td>
</tr>
</tbody>
</table>
Customizing Multicast Routing

This section describes how to customize multicast routing and includes the following topics:

- Configuring Stub Multicast Routing and Forwarding IGMP Messages, page 24-4
- Configuring a Static Multicast Route, page 24-4
- Configuring IGMP Features, page 24-5
- Configuring PIM Features, page 24-9
- Configuring a Bidirectional Neighbor Filter, page 24-13
- Configuring a Multicast Boundary, page 24-14

Configuring Stub Multicast Routing and Forwarding IGMP Messages

Note

Stub multicast routing and PIM are not supported concurrently.

An ASASM acting as the gateway to the stub area does not need to participate in PIM. Instead, you can configure it to act as an IGMP proxy agent and forward IGMP messages from hosts connected on one interface to an upstream multicast router on another interface. To configure the ASASM as an IGMP proxy agent, forward the host join and leave messages from the stub area interface to an upstream interface.

To forward the host join and leave messages, enter the following command from the interface attached to the stub area:

```
igmp forward interface if_name
```

Example:
```
hostname(config-if)# igmp forward interface interface1
```

Configuring a Static Multicast Route

Configuring static multicast routes lets you separate multicast traffic from unicast traffic. For example, when a path between a source and destination does not support multicast routing, the solution is to configure two multicast devices with a GRE tunnel between them and to send the multicast packets over the tunnel.

When using PIM, the ASASM expects to receive packets on the same interface where it sends unicast packets back to the source. In some cases, such as bypassing a route that does not support multicast routing, you may want unicast packets to take one path and multicast packets to take another.

Static multicast routes are not advertised or redistributed.

To configure a static multicast route or a static multicast route for a stub area, enter one of the following commands:
### Configuring IGMP Features

IP hosts use the Internet Group Management Protocol (IGMP) to report their group memberships to directly connected multicast routers.

IGMP is used to dynamically register individual hosts in a multicast group on a particular LAN. Hosts identify group memberships by sending IGMP messages to their local multicast router. Under IGMP, routers listen to IGMP messages and periodically send out queries to discover which groups are active or inactive on a particular subnet.

IGMP uses group addresses (Class D IP address) as group identifiers. Host group address can be in the range of 224.0.0.0 to 239.255.255.255. The address 224.0.0.0 is never assigned to any group. The address 224.0.0.1 is assigned to all systems on a subnet. The address 224.0.0.2 is assigned to all routers on a subnet.

When you enable multicast routing on the ASASM, IGMP Version 2 is automatically enabled on all interfaces.

---

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>**mroute src_ip src_mask {input_if_name</td>
<td>rpf_neighbor} [distance]**</td>
</tr>
</tbody>
</table>

**Example:**

```
hostname(config)# mroute src_ip src_mask {input_if_name | rpf_neighbor} [distance]
```

| mroute src_ip src_mask input_if_name [dense output_if_name] [distance] | Configures a static multicast route for a stub area. The **dense output_if_name** keyword and argument pair is only supported for stub multicast routing. |

**Example:**

```
hostname(config)# mroute src_ip src_mask input_if_name [dense output_if_name] [distance]
```

---

This section describes how to configure optional IGMP setting on a per-interface basis and includes the following topics:

- Disabling IGMP on an Interface, page 24-6
- Configuring IGMP Group Membership, page 24-6
- Configuring a Statically Joined IGMP Group, page 24-6
- Controlling Access to Multicast Groups, page 24-7
- Limiting the Number of IGMP States on an Interface, page 24-7
- Modifying the Query Messages to Multicast Groups, page 24-8
- Changing the IGMP Version, page 24-9

---

**Note**

Only the **no igmp** command appears in the interface configuration when you use the show run command. If the **multicast-routing** command appears in the device configuration, then IGMP is automatically enabled on all interfaces.
Disabling IGMP on an Interface

You can disable IGMP on specific interfaces. This information is useful if you know that there are no multicast hosts on a specific interface and you want to prevent the ASASM from sending host query messages on that interface.

To disable IGMP on an interface, enter the following command:

```
no igmp
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>no igmp</td>
<td>Disables IGMP on an interface.</td>
</tr>
<tr>
<td></td>
<td>To reenable IGMP on an interface, use the <code>igmp</code> command.</td>
</tr>
</tbody>
</table>

Example:

```
hostname(config-if)# no igmp
```

Note: Only the `no igmp` command appears in the interface configuration.

Configuring IGMP Group Membership

You can configure the ASASM to be a member of a multicast group. Configuring the ASASM to join a multicast group causes upstream routers to maintain multicast routing table information for that group and keep the paths for that group active.

Note: If you want to forward multicast packets for a specific group to an interface without the ASASM accepting those packets as part of the group, see the “Configuring a Statically Joined IGMP Group” section on page 24-6.

To have the ASASM join a multicast group, enter the following command:

```
igmp join-group group-address
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>igmp join-group</td>
<td>Configures the ASASM to be a member of a multicast group.</td>
</tr>
<tr>
<td>group-address</td>
<td>The <code>group-address</code> argument is the IP address of the group.</td>
</tr>
</tbody>
</table>

Example:

```
hostname(config-if)# igmp join-group mcast-group
```

Configuring a Statically Joined IGMP Group

Sometimes a group member cannot report its membership in the group because of some configuration, or there may be no members of a group on the network segment. However, you still want multicast traffic for that group to be sent to that network segment. You can have multicast traffic for that group sent to the segment by configuring a statically joined IGMP group.

Enter the `igmp static-group` command. The ASASM does not accept the multicast packets, but instead forwards them to the specified interface.

To configure a statically joined multicast group on an interface, enter the following command:
Chapter 24  Configuring Multicast Routing

Customizing Multicast Routing

To control the multicast groups that hosts on the ASASM interface can join, perform the following steps:

### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td>Do one of the following to create a standard or extended access list:</td>
<td></td>
</tr>
<tr>
<td><strong>access-list</strong> name standard [permit</td>
<td>Creates a standard access list for the multicast traffic.</td>
</tr>
<tr>
<td>deny] ip_addr mask</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# access-list acl1</td>
<td></td>
</tr>
<tr>
<td>standard permit 192.52.662.25</td>
<td></td>
</tr>
<tr>
<td><strong>access-list</strong> name extended [permit</td>
<td>Creates an extended access list.</td>
</tr>
<tr>
<td>deny] protocol src_ip_addr src_mask</td>
<td></td>
</tr>
<tr>
<td>dst_ip_addr dst_mask</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# access-list acl2</td>
<td></td>
</tr>
<tr>
<td>extended permit protocol src_ip_addr</td>
<td></td>
</tr>
<tr>
<td>src_mask dst_ip_addr dst_mask</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td>igmp access-group acl</td>
<td>Applies the access list to an interface.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config-if)# igmp access-group</td>
<td>The acl argument is the name of a standard or extended IP access</td>
</tr>
<tr>
<td>acl</td>
<td>list.</td>
</tr>
</tbody>
</table>

### Limiting the Number of IGMP States on an Interface

You can limit the number of IGMP states resulting from IGMP membership reports on a per-interface basis. Membership reports exceeding the configured limits are not entered in the IGMP cache, and traffic for the excess membership reports is not forwarded.

To limit the number of IGMP states on an interface, enter the following command:
Customizing Multicast Routing

Modifying the Query Messages to Multicast Groups

**Note**
The `igmp query-timeout` and `igmp query-interval` commands require IGMP Version 2.

The ASASM sends query messages to discover which multicast groups have members on the networks attached to the interfaces. Members respond with IGMP report messages indicating that they want to receive multicast packets for specific groups. Query messages are addressed to the all-systems multicast group, which has an address of 224.0.0.1, with a time-to-live value of 1.

These messages are sent periodically to refresh the membership information stored on the ASASM. If the ASASM discovers that there are no local members of a multicast group still attached to an interface, it stops forwarding multicast packet for that group to the attached network, and it sends a prune message back to the source of the packets.

By default, the PIM designated router on the subnet is responsible for sending the query messages. By default, they are sent once every 125 seconds.

When changing the query response time, by default, the maximum query response time advertised in IGMP queries is 10 seconds. If the ASASM does not receive a response to a host query within this amount of time, it deletes the group.

To change the query interval, query response time, and query timeout value, perform the following steps:

**Detailed Steps**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>igmp query-interval seconds</code></td>
<td>Sets the query interval time in seconds. Valid values range from 0 to 500; 125 is the default value.</td>
</tr>
</tbody>
</table>

**Example:**

```
hostname(config-if)# igmp query-interval 30
```

If the ASASM does not hear a query message on an interface for the specified timeout value (by default, 255 seconds), then the ASASM becomes the designated router and starts sending the query messages.
Chapter 24  Configuring Multicast Routing

Customizing Multicast Routing

Changing the IGMP Version

By default, the ASASM runs IGMP Version 2, which enables several additional features such as the `igmp query-timeout` and `igmp query-interval` commands.

All multicast routers on a subnet must support the same version of IGMP. The ASASM does not automatically detect Version 1 routers and switch to Version 1. However, a mix of IGMP Version 1 and 2 hosts on the subnet works; the ASASM running IGMP Version 2 works correctly when IGMP Version 1 hosts are present.

To control which version of IGMP is running on an interface, enter the following command:

```
Command          Purpose
igmp version {1 | 2} Controls the version of IGMP that you want to run on the interface.
```

Example:
```
hostname(config-if)# igmp version 2
```

Configuring PIM Features

Routers use PIM to maintain forwarding tables for forwarding multicast diagrams. When you enable multicast routing on the ASASM, PIM and IGMP are automatically enabled on all interfaces.

Note: PIM is not supported with PAT. The PIM protocol does not use ports, and PAT only works with protocols that use ports.

This section describes how to configure optional PIM settings and includes the following topics:

- Enabling and Disabling PIM on an Interface, page 24-10
- Configuring a Static Rendezvous Point Address, page 24-10
- Configuring the Designated Router Priority, page 24-11
- Configuring and Filtering PIM Register Messages, page 24-11
- Configuring PIM Message Intervals, page 24-12
- Filtering PIM Neighbors, page 24-12
Enabling and Disabling PIM on an Interface

You can enable or disable PIM on specific interfaces. To enable or disable PIM on an interface, perform the following steps:

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 pim</td>
<td>Enables or reenables PIM on a specific interface.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config-if)# pim</td>
<td></td>
</tr>
<tr>
<td>Step 2 no pim</td>
<td>Disables PIM on a specific interface.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config-if)# no pim</td>
<td></td>
</tr>
</tbody>
</table>

Note Only the no pim command appears in the interface configuration.

Configuring a Static Rendezvous Point Address

All routers within a common PIM sparse mode or bidir domain require knowledge of the PIM RP address. The address is statically configured using the pim rp-address command.

Note The ASASM does not support Auto-RP or PIM BSR. You must use the pim rp-address command to specify the RP address.

You can configure the ASASM to serve as RP to more than one group. The group range specified in the access list determines the PIM RP group mapping. If an access list is not specified, then the RP for the group is applied to the entire multicast group range (224.0.0.0/4).

To configure the address of the PIM PR, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>pim rp-address ip_address [acl] [bidir]</td>
<td>Enables or reenables PIM on a specific interface.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# pim rp-address 10.86.75.23 [acl] [bidir]</td>
<td>The ip_address argument is the unicast IP address of the router assigned to be a PIM RP. The acl argument is the name or number of a standard access list that defines with which multicast groups the RP should be used. Do not use a host ACL with this command. Excluding the bidir keyword causes the groups to operate in PIM sparse mode.</td>
</tr>
</tbody>
</table>
The ASASM always advertises the bidirectional capability in the PIM hello messages, regardless of the actual bidirectional configuration.

### Configuring the Designated Router Priority

The DR is responsible for sending PIM register, join, and prune messages to the RP. When there is more than one multicast router on a network segment, selecting the DR is based on the DR priority. If multiple devices have the same DR priority, then the device with the highest IP address becomes the DR.

By default, the ASASM has a DR priority of 1. To change this value, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>pim dr-priority num</code></td>
<td>Changes the designated router priority. The <code>num</code> argument can be any number ranging from 1 to 4294967294.</td>
</tr>
</tbody>
</table>

**Example:**

```
hostname(config-if)# pim dr-priority 500
```

### Configuring and Filtering PIM Register Messages

When the ASASM is acting as an RP, you can restrict specific multicast sources from registering with it to prevent unauthorized sources from registering with the RP. The Request Filter pane lets you define the multicast sources from which the ASASM will accept PIM register messages.

To filter PIM register messages, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`pim accept-register (list acl</td>
<td>route-map map-name)`</td>
</tr>
</tbody>
</table>

**Example:**

```
hostname(config)# pim accept-register (list acl1 | route-map map2)
```
Configuring PIM Message Intervals

Router query messages are used to select the PIM DR. The PIM DR is responsible for sending router query messages. By default, router query messages are sent every 30 seconds. Additionally, every 60 seconds, the ASASM sends PIM join or prune messages.

To change these intervals, perform the following steps:

### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td><code>pim hello-interval seconds</code></td>
<td>Sends router query messages. Valid values for the <code>seconds</code> argument range from 1 to 3600 seconds.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config-if)# pim hello-interval 60</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td><code>pim join-prune-interval seconds</code></td>
<td>Changes the amount of time (in seconds) that the ASASM sends PIM join or prune messages. Valid values for the <code>seconds</code> argument range from 10 to 600 seconds.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config-if)# pim join-prune-interval 60</td>
<td></td>
</tr>
</tbody>
</table>

Filtering PIM Neighbors

You can define the routers that can become PIM neighbors. By filtering the routers that can become PIM neighbors, you can do the following:

- Prevent unauthorized routers from becoming PIM neighbors.
- Prevent attached stub routers from participating in PIM.

To define neighbors that can become a PIM neighbor, perform the following steps:

### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td><code>access-list pim_nbr deny router-IP_addr PIM neighbor</code></td>
<td>Uses a standard access list to define the routers that you want to have participate in PIM. In the example, the following access list, when used with the <code>pim neighbor-filter</code> command, prevents the 10.1.1.1 router from becoming a PIM neighbor.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# access-list pim_nbr deny 10.1.1.1 255.255.255.255</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td><code>pim neighbor-filter pim_nbr</code></td>
<td>Filters neighbor routers. In the example, the 10.1.1.1 router is prevented from becoming a PIM neighbor on interface GigabitEthernet0/3.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# interface GigabitEthernet0/3</td>
<td>hostname(config-if)# pim neighbor-filter pim_nbr</td>
</tr>
</tbody>
</table>
## Configuring a Bidirectional Neighbor Filter

The Bidirectional Neighbor Filter pane shows the PIM bidirectional neighbor filters, if any, that are configured on the ASASM. A PIM bidirectional neighbor filter is an ACL that defines the neighbor devices that can participate in the DF election. If a PIM bidirectional neighbor filter is not configured for an interface, then there are no restrictions. If a PIM bidirectional neighbor filter is configured, only those neighbors permitted by the ACL can participate in the DF election process.

When a PIM bidirectional neighbor filter configuration is applied to the ASASM, an ACL appears in the running configuration with the name `interface-name_multicast`, in which the `interface-name` is the name of the interface to which the multicast boundary filter is applied. If an ACL with that name already exists, a number is appended to the name (for example, `inside_multicast_1`). This ACL defines which devices can become PIM neighbors of the ASASM.

Bidirectional PIM allows multicast routers to keep reduced state information. All of the multicast routers in a segment must be bidirectionally enabled for bidir to elect a DF.

The PIM bidirectional neighbor filters enable the transition from a sparse-mode-only network to a bidir network by letting you specify the routers that should participate in the DF election, while still allowing all routers to participate in the sparse-mode domain. The bidir-enabled routers can elect a DF from among themselves, even when there are non-bidir routers on the segment. Multicast boundaries on the non-bidir routers prevent PIM messages and data from the bidir groups from leaking in or out of the bidir subset cloud.

When a PIM bidirectional neighbor filter is enabled, the routers that are permitted by the ACL are considered to be bidirectionally capable. Therefore, the following is true:

- If a permitted neighbor does not support bidir, then the DF election does not occur.
- If a denied neighbor supports bidir, then the DF election does not occur.
- If a denied neighbor does not support bidir, the DF election can occur.

To define the neighbors that can become a PIM bidirectional neighbor filter, perform the following steps:

### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td><code>access-list pim_nbr deny router-IP_addr</code></td>
<td>Uses a standard access list to define the routers that you want to have participate in PIM.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>In the example, the following access list, when used with the <code>pim neighbor-filter</code> command, prevents the 10.1.1.1 router from becoming a PIM neighbor.</td>
</tr>
<tr>
<td><code>hostname(config)# access-list pim_nbr deny 10.1.1.1 255.255.255.255</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Filters neighbor routers.</td>
</tr>
<tr>
<td><code>pim bidirectional-neighbor-filter pim_nbr</code></td>
<td>In the example, the 10.1.1.1 router is prevented from becoming a PIM bidirectional neighbor on interface GigabitEthernet0/3.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>hostname(config)# interface GigabitEthernet0/3</code></td>
<td></td>
</tr>
<tr>
<td><code>hostname(config-if)# pim bidirectional-neighbor-filter pim_nbr</code></td>
<td></td>
</tr>
</tbody>
</table>
Configuring a Multicast Boundary

Address scoping defines domain boundaries so that domains with RPs that have the same IP address do not leak into each other. Scoping is performed on the subnet boundaries within large domains and on the boundaries between the domain and the Internet.

You can set up an administratively scoped boundary on an interface for multicast group addresses by entering the `multicast boundary` command. IANA has designated the multicast address range from 239.0.0.0 to 239.255.255.255 as the administratively scoped addresses. This range of addresses can be reused in domains administered by different organizations. The addresses would be considered local, not globally unique.

A standard ACL defines the range of affected addresses. When a boundary is set up, no multicast data packets are allowed to flow across the boundary from either direction. The boundary allows the same multicast group address to be reused in different administrative domains.

You can configure, examine, and filter Auto-RP discovery and announcement messages at the administratively scoped boundary by entering the `filter-autorp` keyword. Any Auto-RP group range announcements from the Auto-RP packets that are denied by the boundary ACL are removed. An Auto-RP group range announcement is permitted and passed by the boundary only if all addresses in the Auto-RP group range are permitted by the boundary ACL. If any address is not permitted, the entire group range is filtered and removed from the Auto-RP message before the Auto-RP message is forwarded.

To configure a multicast boundary, enter the following command:

```
hostname(config-if)# multicast boundary acl [filter-autorp]
```

### Command Purpose

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>multicast boundary acl [filter-autorp]</code></td>
<td>Configures a multicast boundary.</td>
</tr>
</tbody>
</table>

**Example:**

```
hostname(config-if)# multicast boundary acl1 [filter-autorp]
```

---

**Configuration Example for Multicast Routing**

The following example shows how to enable and configure multicast routing with various optional processes:

**Step 1** Enable multicast routing:

```
hostname(config)# multicast-routing
```

**Step 2** Configure a static multicast route:

```
hostname(config)# mroute src_ip src_mask (input_if_name | rpf_neighbor) [distance]
hostname(config)# exit
```

**Step 3** Configure the ASASM to be a member of a multicast group:

```
hostname(config)# interface
hostname(config-if)# igmp join-group group-address
```
Additional References

For additional information related to routing, see the following sections:

- Related Documents, page 24-15
- RFCs, page 24-15

Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Technical details about the IGMP and multicast routing standards used for implementing the SMR feature</td>
<td>IETF draft-ietf-idmr-igmp-proxy-01.txt</td>
</tr>
</tbody>
</table>

RFCs

<table>
<thead>
<tr>
<th>RFC</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFC 2113</td>
<td>IP Router Alert Option</td>
</tr>
<tr>
<td>RFC 2236</td>
<td>IGMPv2</td>
</tr>
<tr>
<td>RFC 2362</td>
<td>PIM-SM</td>
</tr>
<tr>
<td>RFC 2588</td>
<td>IP Multicast and Firewalls</td>
</tr>
</tbody>
</table>

Feature History for Multicast Routing

Table 24-2 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multicast routing support</td>
<td>7.0(1)</td>
<td>Support was added for multicast routing data, authentication, and redistribution and monitoring of routing information using the multicast routing protocol. We introduced the <code>multicast-routing</code> command.</td>
</tr>
</tbody>
</table>
CHAPTER 25

Configuring EIGRP

This chapter describes how to configure the ASASM to route data, perform authentication, and redistribute routing information using the Enhanced Interior Gateway Routing Protocol (EIGRP). This chapter includes the following sections:

- Information About EIGRP, page 25-1
- Licensing Requirements for EIGRP, page 25-2
- Guidelines and Limitations, page 25-2
- Configuring EIGRP, page 25-3
- Customizing EIGRP, page 25-4
- Monitoring EIGRP, page 25-17
- Configuration Example for EIGRP, page 25-18
- Feature History for EIGRP, page 25-19

Information About EIGRP

EIGRP is an enhanced version of IGRP developed by Cisco. Unlike IGRP and RIP, EIGRP does not send out periodic route updates. EIGRP updates are sent out only when the network topology changes. Key capabilities that distinguish EIGRP from other routing protocols include fast convergence, support for variable-length subnet mask, support for partial updates, and support for multiple network layer protocols.

A router running EIGRP stores all the neighbor routing tables so that it can quickly adapt to alternate routes. If no appropriate route exists, EIGRP queries its neighbors to discover an alternate route. These queries propagate until an alternate route is found. Its support for variable-length subnet masks permits routes to be automatically summarized on a network number boundary. In addition, EIGRP can be configured to summarize on any bit boundary at any interface. EIGRP does not make periodic updates. Instead, it sends partial updates only when the metric for a route changes. Propagation of partial updates is automatically bounded so that only those routers that need the information are updated. As a result of these two capabilities, EIGRP consumes significantly less bandwidth than IGRP.

Neighbor discovery is the process that the ASASM uses to dynamically learn of other routers on directly attached networks. EIGRP routers send out multicast hello packets to announce their presence on the network. When the ASASM receives a hello packet from a new neighbor, it sends its topology table to the neighbor with an initialization bit set. When the neighbor receives the topology update with the initialization bit set, the neighbor sends its topology table back to the ASASM.
The hello packets are sent out as multicast messages. No response is expected to a hello message. The exception to this is for statically defined neighbors. If you use the `neighbor` command, or configure the Hello Interval in ASDM, to configure a neighbor, the hello messages sent to that neighbor are sent as unicast messages. Routing updates and acknowledgements are sent out as unicast messages.

Once this neighbor relationship is established, routing updates are not exchanged unless there is a change in the network topology. The neighbor relationship is maintained through the hello packets. Each hello packet received from a neighbor includes a hold time. This is the time in which the ASASM can expect to receive a hello packet from that neighbor. If the ASASM does not receive a hello packet from that neighbor within the hold time advertised by that neighbor, the ASASM considers that neighbor to be unavailable.

The EIGRP protocol uses four key algorithm technologies, including neighbor discovery/recovery, Reliable Transport Protocol (RTP), and DUAL, which is important for route computations. DUAL saves all routes to a destination in the topology table, not just the least-cost route. The least-cost route is inserted into the routing table. The other routes remain in the topology table. If the main route fails, another route is chosen from the feasible successors. A successor is a neighboring router used for packet forwarding that has a least-cost path to a destination. The feasibility calculation guarantees that the path is not part of a routing loop.

If a feasible successor is not found in the topology table, a route recomputation must occur. During route recomputation, DUAL queries the EIGRP neighbors for a route, who in turn query their neighbors. Routers that do not have a feasible successor for the route return an unreachable message.

During route recomputation, DUAL marks the route as active. By default, the ASASM waits for three minutes to receive a response from its neighbors. If the ASASM does not receive a response from a neighbor, the route is marked as stuck-in-active. All routes in the topology table that point to the unresponsive neighbor as a feasibility successor are removed.

---

**Note**

EIGRP neighbor relationships are not supported through the IPsec tunnel without a GRE tunnel.

---

### Licensing Requirements for EIGRP

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

---

### Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

**Context Mode Guidelines**

Supported in single context mode.

**Firewall Mode Guidelines**

Supported only in routed firewall mode. Transparent firewall mode is not supported.
IPv6 Guidelines
Does not support IPv6.

Configuring EIGRP

This section describes how to enable the EIGRP process on your system. After you have enabled EIGRP, see the following sections to learn how to customize the EIGRP process on your system.

- Enabling EIGRP, page 25-3
- Enabling EIGRP Stub Routing, page 25-3

Enabling EIGRP

You can only enable one EIGRP routing process on the ASASM.

To enable EIGRP, perform the following steps:

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td></td>
</tr>
<tr>
<td>router eigrp as-num</td>
<td>Creates an EIGRP routing process and enters router configuration mode for this EIGRP process. The as-num argument is the autonomous system number of the EIGRP routing process.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# router eigrp 2</td>
<td></td>
</tr>
<tr>
<td>Step 2</td>
<td></td>
</tr>
<tr>
<td>network ip-addr [mask]</td>
<td>Configures the interfaces and networks that participate in EIGRP routing. You can configure one or more network statements with this command. Directly connected and static networks that fall within the defined network are advertised by the ASASM. Additionally, only interfaces with an IP address that fall within the defined network participate in the EIGRP routing process. If you have an interface that you do not want to have participate in EIGRP routing, but that is attached to a network that you want advertised, see the “Configuring Interfaces for EIGRP” section on page 25-6.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# router eigrp 2</td>
<td></td>
</tr>
<tr>
<td>hostname(config-router)# network 10.0.0.0 255.0.0.0</td>
<td></td>
</tr>
</tbody>
</table>

Enabling EIGRP Stub Routing

You can enable, and configure the ASASM as an EIGRP stub router. Stub routing decreases memory and processing requirements on the ASASM. As a stub router, the ASASM does not need to maintain a complete EIGRP routing table because it forwards all nonlocal traffic to a distribution router. Generally, the distribution router need not send anything more than a default route to the stub router.

Only specified routes are propagated from the stub router to the distribution router. As a stub router, the ASASM responds to all queries for summaries, connected routes, redistributed static routes, external routes, and internal routes with the message “inaccessible.” When the ASASM is configured as a stub, it sends a special peer information packet to all neighboring routers to report its status as a stub router.
Any neighbor that receives a packet informing it of the stub status will not query the stub router for any routes, and a router that has a stub peer will not query that peer. The stub router depends on the distribution router to send the correct updates to all peers.

To enable the ASASM as an EIGRP stub routing process, perform the following steps:

### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> router eigrp as-num</td>
<td>Creates an EIGRP routing process and enters router configuration mode for this EIGRP process. The <code>as-num</code> argument is the autonomous system number of the EIGRP routing process.</td>
</tr>
<tr>
<td><strong>Example:</strong> hostname(config)# router eigrp 2</td>
<td></td>
</tr>
</tbody>
</table>

| **Step 2** network ip-addr [mask] | Configures the interfaces and networks that participate in EIGRP routing. You can configure one or more `network` statements with this command. Directly connected and static networks that fall within the defined network are advertised by the ASASM. Additionally, only interfaces with an IP address that fall within the defined network participate in the EIGRP routing process. If you have an interface that you do not want to have participate in EIGRP routing, but that is attached to a network that you want advertised, see the section “Configuring Passive Interfaces” section on page 25-7. | |
| **Example:** hostname(config)# router eigrp 2 hostname(config-router)# network 10.0.0.0 255.0.0.0 | |

| **Step 3** eigrp stub {receive-only | [connected] | [redistributed] | [static] | [summary]} | Configures the stub routing process. You must specify which networks are advertised by the stub routing process to the distribution router. Static and connected networks are not automatically redistributed into the stub routing process. | |
| **Example:** hostname(config)# router eigrp 2 hostname(config-router)# network 10.0.0.0 255.0.0.0 hostname(config-router)# eigrp stub {receive-only | [connected] | [redistributed] | [static] | [summary]} | |

**Note**

A stub routing process does not maintain a full topology table. At a minimum, stub routing needs a default route to a distribution router, which makes the routing decisions.

### Customizing EIGRP

This section describes how to customize the EIGRP routing and includes the following topics:

- Defining a Network for an EIGRP Routing Process, page 25-5
- Configuring Interfaces for EIGRP, page 25-6
- Configuring the Summary Aggregate Addresses on Interfaces, page 25-8
- Changing the Interface Delay Value, page 25-9
Defining a Network for an EIGRP Routing Process

The Network table lets you specify the networks used by the EIGRP routing process. For an interface to participate in EIGRP routing, it must fall within the range of addresses defined by the network entries. For directly connected and static networks to be advertised, they must also fall within the range of the network entries.

The Network table displays the networks configured for the EIGRP routing process. Each row of the table displays the network address and associated mask configured for the specified EIGRP routing process.

To add or define a network, perform the following steps:

### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>router eigrp as-num</code></td>
<td>Creates an EIGRP routing process and enters router configuration mode for this EIGRP process. The <code>as-num</code> argument is the autonomous system number of the EIGRP routing process.</td>
</tr>
<tr>
<td>Example: <code>hostname(config)# router eigrp 2</code></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>network ip-addr [mask]</code></td>
<td>Configures the interfaces and networks that participate in EIGRP routing. You can configure one or more <code>network</code> statements with this command. Directly connected and static networks that fall within the defined network are advertised by the ASASM. Additionally, only interfaces with an IP address that fall within the defined network participate in the EIGRP routing process. If you have an interface that you do not want to have participate in EIGRP routing, but that is attached to a network that you want advertised, see the “Configuring Passive Interfaces” section on page 25-7.</td>
</tr>
</tbody>
</table>
| Example: `hostname(config)# router eigrp 2
hostname(config-router)# network 10.0.0.0 255.0.0.0` | |
Configuring Interfaces for EIGRP

If you have an interface that you do not want to have participate in EIGRP routing, but that is attached to a network that you want advertised, you can configure a network command that includes the network to which the interface is attached, and use the passive-interface command to prevent that interface from sending or receiving EIGRP updates.

To configure interfaces for EIGRP, perform the following steps:

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1**
router eigrp as-num | Creates an EIGRP routing process and enters router configuration mode for this EIGRP process.
The as-num argument is the autonomous system number of the EIGRP routing process. |
| Example:
hostname(config)# router eigrp 2 | Configures the interfaces and networks that participate in EIGRP routing. You can configure one or more network statements with this command. |
| **Step 2**
network ip-addr [mask] | Directly connected and static networks that fall within the defined network are advertised by the ASASM. Additionally, only interfaces with an IP address that fall within the defined network participate in the EIGRP routing process. |
| Example:
hostname(config)# router eigrp 2
hostname(config-router)# network 10.0.0.0 255.0.0.0 | If you have an interface that you do not want to have participate in EIGRP routing, but that is attached to a network that you want advertised, see the “Defining a Network for an EIGRP Routing Process” section on page 25-5. |
| **Step 3**
(Optional) Do one of the following to customize an interface to participate in EIGRP routing:

no default-information {in | out | WORD} | Allows you to control the sending or receiving of candidate default route information. |
| Example:
hostname(config)# router eigrp 2
hostname(config-router)# network 10.0.0.0 255.0.0.0
hostname(config-router)# no default-information {in | out | WORD} | Enter the no default-information in command causes the candidate default route bit to be blocked on received routes. |
| authentication mode eigrp as-num md5 | Enables MD5 authentication of EIGRP packets. |
| Example:
hostname(config)# authentication mode eigrp 2 md5 | The as-num argument is the autonomous system number of the EIGRP routing process configured on the ASASM. If EIGRP is not enabled or if you enter the wrong number, the ASASM returns the following error message: |
% Asystem(100) specified does not exist | See the “Enabling EIGRP Authentication on an Interface” section on page 25-9 for more information on this particular option. |
Configuring Passive Interfaces

You can configure one or more interfaces as passive interfaces. In EIGRP, a passive interface does not send or receive routing updates.

To configure passive interfaces, perform the following steps:

### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>router eigrp as-num</td>
<td>Creates an EIGRP routing process and enters router configuration mode for this EIGRP process. The <code>as-num</code> argument is the autonomous system number of the EIGRP routing process.</td>
</tr>
</tbody>
</table>
Customizing EIGRP

Configuring the Summary Aggregate Addresses on Interfaces

You can configure a summary addresses on a per-interface basis. You need to manually define summary addresses if you want to create summary addresses that do not occur at a network number boundary or if you want to use summary addresses on an ASASM with automatic route summarization disabled. If any more specific routes are in the routing table, EIGRP will advertise the summary address out the interface with a metric equal to the minimum of all more specific routes.

To create a summary address, perform the following steps:

**Detailed Steps**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> interface phy_if</td>
<td>Enters interface configuration mode for the interface on which you are changing the delay value used by EIGRP.</td>
</tr>
<tr>
<td><strong>Example:</strong> hostname(config)# interface phy_if</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> summary-address eigrp as-num address mask [distance]</td>
<td>Creates the summary address. By default, EIGRP summary addresses that you define have an administrative distance of 5. You can change this value by specifying the optional distance argument in the summary-address command.</td>
</tr>
<tr>
<td><strong>Example:</strong> hostname(config-if)# summary-address eigrp 2 address mask [20]</td>
<td></td>
</tr>
</tbody>
</table>
## Changing the Interface Delay Value

The interface delay value is used in EIGRP distance calculations. You can modify this value on a per-interface basis.

To change the interface delay value, perform the following steps:

### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><strong>interface phy_if</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><strong>hostname(config)# interface phy_if</strong></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><strong>delay value</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><strong>hostname(config-if)# delay 200</strong></td>
</tr>
</tbody>
</table>

### Enabling EIGRP Authentication on an Interface

EIGRP route authentication provides MD5 authentication of routing updates from the EIGRP routing protocol. The MD5 keyed digest in each EIGRP packet prevents the introduction of unauthorized or false routing messages from unapproved sources.

EIGRP route authentication is configured on a per-interface basis. All EIGRP neighbors on interfaces configured for EIGRP message authentication must be configured with the same authentication mode and key for adjacencies to be established.

**Note**

Before you can enable EIGRP route authentication, you must enable EIGRP.

To enable EIGRP authentication on an interface, perform the following steps:
### Detailed Steps

**Step 1**  
```
router eigrp as-num
```

**Example:**
```
hostname(config)# router eigrp 2
```

Creates an EIGRP routing process and enters router configuration mode for this EIGRP process. The `as-num` argument is the autonomous system number of the EIGRP routing process.

**Step 2**  
```
network ip-addr [mask]
```

**Example:**
```
hostname(config)# router eigrp 2
hostname(config-router)# network 10.0.0.0 255.0.0.0
```

Configures the interfaces and networks that participate in EIGRP routing. You can configure one or more network statements with this command. Directly connected and static networks that fall within the defined network are advertised by the ASASM. Additionally, only interfaces with an IP address that falls within the defined network participate in the EIGRP routing process.

If you have an interface that you do not want to have participate in EIGRP routing, but that is attached to a network that you want advertised, see the “Configuring EIGRP” section on page 25-3.

**Step 3**  
```
interface phy_if
```

**Example:**
```
hostname(config)# interface phy_if
```

Enters interface configuration mode for the interface on which you are configuring EIGRP message authentication.

**Step 4**  
```
authentication mode eigrp as-num md5
```

**Example:**
```
hostname(config)# authentication mode eigrp 2 md5
```

Enables MD5 authentication of EIGRP packets. The `as-num` argument is the autonomous system number of the EIGRP routing process configured on the ASASM. If EIGRP is not enabled or if you enter the wrong number, the ASASM returns the following error message:

```
% Asystem(100) specified does not exist
```

**Step 5**  
```
authentication key eigrp as-num key key-id
```

**Example:**
```
hostname(config)# authentication key eigrp 2 cisco key-id 200
```

Configures the key used by the MD5 algorithm. The `as-num` argument is the autonomous system number of the EIGRP routing process configured on the ASASM. If EIGRP is not enabled or if you enter the wrong number, the ASASM returns the following error message:

```
% Asystem(100) specified does not exist
```

The `key` argument can include up to 16 characters. The `key-id` argument is a number that can range from 0 to 255.

---

### Defining an EIGRP Neighbor

EIGRP hello packets are sent as multicast packets. If an EIGRP neighbor is located across a non broadcast network, such as a tunnel, you must manually define that neighbor. When you manually define an EIGRP neighbor, hello packets are sent to that neighbor as unicast messages.

To manually define an EIGRP neighbor, perform the following steps:
### Customizing EIGRP

#### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><strong>router eigrp as-num</strong>&lt;br&gt;Example: hostname(config)# router eigrp 2</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><strong>neighbor ip-addr interface if_name</strong>&lt;br&gt;Example: hostname(config)# router eigrp 2 hostname(config-router)# neighbor 10.0.0.0 interface interface1</td>
</tr>
</tbody>
</table>

#### Redistributing Routes Into EIGRP

You can redistribute routes discovered by RIP and OSPF into the EIGRP routing process. You can also redistribute static and connected routes into the EIGRP routing process. You do not need to redistribute connected routes if they fall within the range of a `network` statement in the EIGRP configuration.

Note: For RIP only: Before you begin this procedure, you must create a route-map to further define which routes from the specified routing protocol are redistributed into the RIP routing process. See Chapter 21, “Defining Route Maps,” for more information about creating a route map.

To redistribute routes into the EIGRP routing process, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><strong>router eigrp as-num</strong>&lt;br&gt;Example: hostname(config)# router eigrp 2</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><strong>default-metric</strong>&lt;br&gt;Example: hostname(config)# router eigrp 2 hostname(config-router)# default-metric bandwidth delay reliability loading mtu</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>Do one of the following to redistribute the selected route type into the EIGRP routing process. You must specify the EIGRP metric values in the <code>redistribute</code> command if you do not have a <code>default-metric</code> command in the EIGRP router configuration.</td>
</tr>
</tbody>
</table>
### Customizing EIGRP

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>redistribute connected [metric bandwidth delay reliability loading mtu] [route-map map_name]</td>
<td>Redistributes connected routes into the EIGRP routing process.</td>
</tr>
</tbody>
</table>

**Example:**
```
hostname(config-router): redistribute connected [metric bandwidth delay reliability loading mtu] [route-map map_name]
```

| redistribute static [metric bandwidth delay reliability loading mtu] [route-map map_name] | Redistributes static routes into the EIGRP routing process. |

**Example:**
```
hostname(config-router): redistribute static [metric bandwidth delay reliability loading mtu] [route-map map_name]
```

| redistribute ospf pid [match {internal | external [1 | 2] | nssa-external [1 | 2]}] [metric bandwidth delay reliability loading mtu] [route-map map_name] | Redistributes routes from an OSPF routing process into the EIGRP routing process. |

**Example:**
```
hostname(config-router): redistribute ospf pid [match {internal | external [1 | 2] | nssa-external [1 | 2]}] [metric bandwidth delay reliability loading mtu] [route-map map_name]
```

| redistribute rip [metric bandwidth delay reliability load mtu] [route-map map_name] | Redistributes routes from a RIP routing process into the EIGRP routing process. |

**Example:**
```
(config-router): redistribute rip [metric bandwidth delay reliability load mtu] [route-map map_name]
```

### Filtering Networks in EIGRP

**Note**
Before you begin this process, you must create a standard access list that defines the routes that you want to advertise. That is, create a standard access list that defines the routes that you want to filter from sending or receiving updates.
To filter networks in EIGRP, perform the following steps:

**Detailed Steps**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><strong>router eigrp as-num</strong>&lt;br&gt;&lt;br&gt;Example:&lt;br&gt;hostname(config)# router eigrp 2</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><strong>hostname(config-router)# network ip-addr [mask]</strong>&lt;br&gt;&lt;br&gt;Example:&lt;br&gt;hostname(config)# router eigrp 2&lt;br&gt;hostname(config-router)# network 10.0.0.0 255.0.0.0</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>Do one of the following to filter networks sent or received in EIGRP routing updates. You can enter multiple <code>distribute-list</code> commands in your EIGRP router configuration.&lt;br&gt;&lt;br&gt;`distribute-list acl out [connected</td>
</tr>
<tr>
<td></td>
<td><code>distribute-list acl in [interface if_name]</code>&lt;br&gt;&lt;br&gt;Example:&lt;br&gt;hostname(config)# router eigrp 2&lt;br&gt;hostname(config-router)# network 10.0.0.0 255.0.0.0&lt;br&gt;hostname(config-router): distribute-list acl in [interface interface1]</td>
</tr>
</tbody>
</table>

**Customizing the EIGRP Hello Interval and Hold Time**

The ASASM periodically sends hello packets to discover neighbors and to learn when neighbors become unreachable or inoperative. By default, hello packets are sent every 5 seconds.

The hello packet advertises the ASASM hold time. The hold time indicates to EIGRP neighbors the length of time the neighbor should consider the ASASM reachable. If the neighbor does not receive a hello packet within the advertised hold time, then the ASASM is considered unreachable. By default, the advertised hold time is 15 seconds (three times the hello interval).
Both the hello interval and the advertised hold time are configured on a per-interface basis. We recommend setting the hold time to be at minimum three times the hello interval.

To configure the hello interval and advertised hold time, perform the following steps:

### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td><code>interface ph_if</code></td>
<td>Enters interface configuration mode for the interface on which you are configuring the hello interval or advertised hold time.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# interface ph_if</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td><code>hello-interval eigrp as-num seconds</code></td>
<td>Changes the hello interval.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# hello-interval eigrp 2 60</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td></td>
</tr>
<tr>
<td><code>hold-time eigrp as-num seconds</code></td>
<td>Changes the hold time.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# hold-time eigrp 2 60</td>
<td></td>
</tr>
</tbody>
</table>

### Disabling Automatic Route Summarization

Automatic route summarization is enabled by default. The EIGRP routing process summarizes on network number boundaries. This can cause routing problems if you have noncontiguous networks.

For example, if you have a router with the networks 192.168.1.0, 192.168.2.0, and 192.168.3.0 connected to it, and those networks all participate in EIGRP, the EIGRP routing process creates the summary address 192.168.0.0 for those routes. If an additional router is added to the network with the networks 192.168.10.0 and 192.168.11.0, and those networks participate in EIGRP, they will also be summarized as 192.168.0.0. To prevent the possibility of traffic being routed to the wrong location, you should disable automatic route summarization on the routers creating the conflicting summary addresses.

To disable automatic route summarization, enter the following commands:
Configuring Default Information in EIGRP

You can control the sending and receiving of default route information in EIGRP updates. By default, default routes are sent and accepted. Configuring the ASASM to disallow default information to be received causes the candidate default route bit to be blocked on received routes. Configuring the ASASM to disallow default information to be sent disables the setting of the default route bit in advertised routes.

To configure default routing information, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td></td>
</tr>
<tr>
<td><code>router eigrp as-num</code></td>
<td>Creates an EIGRP routing process and enters router configuration mode for this EIGRP process.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>hostname(config)# router eigrp 2</code></td>
<td>The <code>as-num</code> argument is the autonomous system number of the EIGRP routing process.</td>
</tr>
<tr>
<td>Step 2</td>
<td></td>
</tr>
<tr>
<td><code>no auto-summary</code></td>
<td>You cannot configure this value. Automatic summary addresses have an administrative distance of 5.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>hostname(config-router)# no auto-summary</code></td>
<td></td>
</tr>
</tbody>
</table>
Customizing EIGRP

Disabling EIGRP Split Horizon

Split horizon controls the sending of EIGRP update and query packets. When split horizon is enabled on an interface, update and query packets are not sent for destinations for which this interface is the next hop. Controlling update and query packets in this manner reduces the possibility of routing loops.

By default, split horizon is enabled on all interfaces.

Split horizon blocks route information from being advertised by a router out of any interface from which that information originated. This behavior usually optimizes communications among multiple routing devices, particularly when links are broken. However, with nonbroadcast networks, there may be situations where this behavior is not desired. For these situations, including networks in which you have EIGRP configured, you may want to disable split horizon.

If you disable split horizon on an interface, you must disable it for all routers and access servers on that interface.
To disable EIGRP split horizon, perform the following steps:

### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> interface phy_if</td>
<td>Enters interface configuration mode for the interface on which you are changing the delay value used by EIGRP.</td>
</tr>
<tr>
<td><strong>Example:</strong> hostname(config)# interface phy_if</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> no split-horizon eigrp as-number</td>
<td>Disables the split horizon.</td>
</tr>
<tr>
<td><strong>Example:</strong> hostname(config-if)# no split-horizon eigrp 2</td>
<td></td>
</tr>
</tbody>
</table>

### Restarting the EIGRP Process

To restart an EIGRP process or clear redistribution or counters, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>clear eigrp pid {1-65535</td>
<td>neighbors</td>
</tr>
<tr>
<td><strong>Example:</strong> hostname(config)# clear eigrp pid 10 neighbors</td>
<td></td>
</tr>
</tbody>
</table>

### Monitoring EIGRP

You can use the following commands to monitor the EIGRP routing process. For examples and descriptions of the command output, see the command reference. Additionally, you can disable the logging of neighbor change messages and neighbor warning messages.

To monitor or disable various EIGRP routing statistics, enter one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Monitoring EIGRP Routing</strong></td>
<td></td>
</tr>
<tr>
<td>show eigrp [as-number] events [{start end]</td>
<td>Displays the EIGRP event log.</td>
</tr>
<tr>
<td>type]</td>
<td></td>
</tr>
<tr>
<td>show eigrp [as-number] neighbors [detail</td>
<td>Displays the EIGRP neighbor table.</td>
</tr>
<tr>
<td>static] [if-name]</td>
<td></td>
</tr>
<tr>
<td>show eigrp [as-number] interfaces [if-name]</td>
<td>Displays the interfaces participating in EIGRP</td>
</tr>
<tr>
<td>[detail]</td>
<td>routing.</td>
</tr>
<tr>
<td>show eigrp [as-number] topology [ip-addr</td>
<td>Displays the EIGRP topology table.</td>
</tr>
<tr>
<td>[mask]</td>
<td>active</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Configuration Example for EIGRP

The following example shows how to enable and configure EIGRP with various optional processes:

### Step 1
To enable EIGRP, enter the following commands:

```
hostname(config)# router eigrp 2
hostname(config-router)# network 10.0.0.0 255.0.0.0
```

### Step 2
To configure an interface from sending or receiving EIGRP routing messages, enter the following command:

```
hostname(config-router)# passive-interface {default}
```

### Step 3
To define an EIGRP neighbor, enter the following command:

```
hostname(config-router)# neighbor 10.0.0.0 interface interface1
```

### Step 4
To configure the interfaces and networks that participate in EIGRP routing, enter the following command:

```
hostname(config-router)# network 10.0.0.0 255.0.0.0
```

### Step 5
To change the interface delay value used in EIGRP distance calculations, enter the following commands:

```
hostname(config-router)# exit
hostname(config)# interface phy_if
hostname(config-if)# delay 200
```

---

<table>
<thead>
<tr>
<th>Command (continued)</th>
<th>Purpose (continued)</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show eigrp [as-number] traffic</code></td>
<td>Displays EIGRP traffic statistics.</td>
</tr>
<tr>
<td><code>router-id</code></td>
<td>Displays the router-id for this EIGRP process.</td>
</tr>
</tbody>
</table>

**Disabling EIGRP Logging Messages**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>no eigrp log-neighbor-changes</code></td>
<td>Disables the logging of neighbor change messages. Enter this command in router configuration mode for the EIGRP routing process.</td>
</tr>
<tr>
<td><code>no eigrp log-neighbor-warnings</code></td>
<td>Disables the logging of neighbor warning messages.</td>
</tr>
</tbody>
</table>

---

**Note**

By default, neighbor change and neighbor warning messages are logged.
## Feature History for EIGRP

Table 25-1 lists each feature change and the platform release in which it was implemented.

### Table 25-1 Feature History for EIGRP

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>EIGRP support</td>
<td>7.0(1)</td>
<td>Support was added for routing data, performing authentication, and redistributing and monitoring routing information using the Enhanced Interior Gateway Routing Protocol (EIGRP).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>We introduced the following command: <code>route eigrp</code>.</td>
</tr>
</tbody>
</table>
Configuring IPv6 Neighbor Discovery

This chapter describes how to enable and configure IPv6 neighbor discovery on the ASASM and includes the following sections:

- Information About IPv6 Neighbor Discovery, page 26-1
- Licensing Requirements for IPv6 Neighbor Discovery, page 26-4
- Guidelines and Limitations, page 26-4
- Default Settings for IPv6 Neighbor Discovery, page 26-6
- Configuring the Neighbor Solicitation Message Interval, page 26-7
- Configuring the Neighbor Reachable Time, page 26-7
- Configuring the Router Advertisement Transmission Interval, page 26-8
- Configuring the Router Lifetime Value, page 26-8
- Configuring DAD Settings, page 26-9
- Configuring IPv6 Addresses on an Interface, page 26-9
- Suppressing Router Advertisement Messages, page 26-10
- Configuring the IPv6 Prefix, page 26-11
- Configuring a Static IPv6 Neighbor, page 26-12
- Monitoring IPv6 Neighbor Discovery, page 26-13
- Additional References, page 26-13
- Feature History for IPv6 Neighbor Discovery, page 26-14

Information About IPv6 Neighbor Discovery

The IPv6 neighbor discovery process uses ICMPv6 messages and solicited-node multicast addresses to determine the link-layer address of a neighbor on the same network (local link), verify the readability of a neighbor, and keep track of neighboring routers.

Nodes (hosts) use neighbor discovery to determine the link-layer addresses for neighbors known to reside on attached links and to quickly purge cached values that become invalid. Hosts also use neighbor discovery to find neighboring routers that are willing to forward packets on their behalf. In addition, nodes use the protocol to actively keep track of which neighbors are reachable and which are not, and to detect changed link-layer addresses. When a router or the path to a router fails, a host actively searches for functioning alternates.
This section includes the following topics:

- Neighbor Solicitation Messages, page 26-2
- Neighbor Reachable Time, page 26-3
- Router Advertisement Messages, page 26-3
- Static IPv6 Neighbors, page 26-4

**Neighbor Solicitation Messages**

Neighbor solicitation messages (ICMPv6 Type 135) are sent on the local link by nodes attempting to discover the link-layer addresses of other nodes on the local link. The neighbor solicitation message is sent to the solicited-node multicast address. The source address in the neighbor solicitation message is the IPv6 address of the node sending the neighbor solicitation message. The neighbor solicitation message also includes the link-layer address of the source node.

After receiving a neighbor solicitation message, the destination node replies by sending a neighbor advertisement message (ICMPv6 Type 136) on the local link. The source address in the neighbor advertisement message is the IPv6 address of the node sending the neighbor advertisement message; the destination address is the IPv6 address of the node that sent the neighbor solicitation message. The data portion of the neighbor advertisement message includes the link-layer address of the node sending the neighbor advertisement message.

After the source node receives the neighbor advertisement, the source node and destination node can communicate. Figure 26-1 shows the neighbor solicitation and response process.

![IPv6 Neighbor Discovery—Neighbor Solicitation Message](image)

Neighbor solicitation messages are also used to verify the reachability of a neighbor after the link-layer address of a neighbor is identified. When a node wants to verifying the reachability of a neighbor, the destination address in a neighbor solicitation message is the unicast address of the neighbor.

Neighbor advertisement messages are also sent when there is a change in the link-layer address of a node on a local link. When there is such a change, the destination address for the neighbor advertisement is the all-nodes multicast address.
Neighbor Reachable Time

The neighbor reachable time enables detecting unavailable neighbors. Shorter configured times enable detecting unavailable neighbors more quickly, however, shorter times consume more IPv6 network bandwidth and processing resources in all IPv6 network devices. Very short configured times are not recommended in normal IPv6 operation.

Router Advertisement Messages

An ASASM can participate in router advertisements so that neighboring devices can dynamically learn a default router address. Router advertisement messages (ICMPv6 Type 134) are periodically sent out each IPv6 configured interface of the ASASM. The router advertisement messages are sent to the all-nodes multicast address. Figure 26-2 shows the router advertisement messages that are sent from IPv6 configured interfaces on the ASASM.

Router advertisement messages typically include the following information:

- One or more IPv6 prefix that nodes on the local link can use to automatically configure their IPv6 addresses.
- Lifetime information for each prefix included in the advertisement.
- Sets of flags that indicate the type of autoconfiguration (stateless or stateful) that can be completed.
- Default router information (whether the router sending the advertisement should be used as a default router and, if so, the amount of time (in seconds) the router should be used as a default router).
- Additional information for hosts, such as the hop limit and MTU a host should use in packets that it originates.
- The amount of time between neighbor solicitation message retransmissions on a given link.
- The amount of time a node considers a neighbor reachable.

Router advertisements are also sent in response to router solicitation messages (ICMPv6 Type 133). Router solicitation messages are sent by hosts at system startup so that the host can immediately autoconfigure without needing to wait for the next scheduled router advertisement message. Because router solicitation messages are usually sent by hosts at system startup and the host does not have a configured unicast address, the source address in router solicitation messages is usually the unspecified IPv6 address (0:0:0:0:0:0:0:0). If the host has a configured unicast address, the unicast address of the interface sending the router solicitation message is used as the source address in the message. The...
destination address in router solicitation messages is the all-routers multicast address with a scope of the link. When a router advertisement is sent in response to a router solicitation, the destination address in the router advertisement message is the unicast address of the source of the router solicitation message.

You can configure the following settings for router advertisement messages:

- The time interval between periodic router advertisement messages.
- The router lifetime value, which indicates the amount of time IPv6 nodes should consider the ASASM to be the default router.
- The IPv6 network prefixes in use on the link.
- Whether or not an interface transmits router advertisement messages.

Unless otherwise noted, the router advertisement message settings are specific to an interface and are entered in interface configuration mode.

**Static IPv6 Neighbors**

You can manually define a neighbor in the IPv6 neighbor cache. If an entry for the specified IPv6 address already exists in the neighbor discovery cache—learned through the IPv6 neighbor discovery process—the entry is automatically converted to a static entry. Static entries in the IPv6 neighbor discovery cache are not modified by the neighbor discovery process.

**Licensing Requirements for IPv6 Neighbor Discovery**

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

**Guidelines and Limitations**

This section includes the guidelines and limitations for this feature.

**Context Mode Guidelines**
Supported in single and multiple context mode.

**Firewall Mode Guidelines**
Supported in routed mode only. Transparent mode is not supported.

**Additional Guidelines and Limitations**

- The interval value is included in all IPv6 router advertisements that are sent out of this interface.
The configured time enables detecting unavailable neighbors. Shorter configured times enable detecting unavailable neighbors more quickly; however, shorter times consume more IPv6 network bandwidth and processing resources in all IPv6 network devices. Very short configured times are not recommended in normal IPv6 operation.

The interval between transmissions should be less than or equal to the IPv6 router advertisement lifetime if the ASASM is configured as a default router by using the `ipv6 nd ra-lifetime` command. To prevent synchronization with other IPv6 nodes, randomly adjust the actual value used to within 20 percent of the specified value.

The `ipv6 nd prefix` command allows control over the individual parameters per prefix, including whether or not the prefix should be advertised.

By default, prefixes configured as addresses on an interface using the `ipv6 address` command are advertised in router advertisements. If you configure prefixes for advertisement using the `ipv6 nd prefix` command, then only these prefixes are advertised.

The `default` keyword can be used to set default parameters for all prefixes.

A date can be set to specify the expiration of a prefix. The valid and preferred lifetimes are counted down in real time. When the expiration date is reached, the prefix will no longer be advertised.

When onlink is on (by default), the specified prefix is assigned to the link. Nodes sending traffic to such addresses that contain the specified prefix consider the destination to be locally reachable on the link.

When autoconfig is on (by default), it indicates to hosts on the local link that the specified prefix can be used for IPv6 autoconfiguration.

For stateless autoconfiguration to work correctly, the advertised prefix length in router advertisement messages must always be 64 bits.

The router lifetime value is included in all IPv6 router advertisements sent out of the interface. The value indicates the usefulness of the ASASM as a default router on this interface.

Setting the value to a non-zero value indicates that the ASASM should be considered a default router on this interface. The non-zero value for the router lifetime value should not be less than the router advertisement interval.

The following guidelines and limitations apply for configuring a static IPv6 neighbor:

- The `ipv6 neighbor` command is similar to the `arp` command. If an entry for the specified IPv6 address already exists in the neighbor discovery cache—learned through the IPv6 neighbor discovery process—the entry is automatically converted to a static entry. These entries are stored in the configuration when the `copy` command is used to store the configuration.

- Use the `show ipv6 neighbor` command to view static entries in the IPv6 neighbor discovery cache.

- The `clear ipv6 neighbor` command deletes all entries in the IPv6 neighbor discovery cache except static entries. The `no ipv6 neighbor` command deletes a specified static entry from the neighbor discovery cache; the command does not remove dynamic entries—entries learned from the IPv6 neighbor discovery process—from the cache. Disabling IPv6 on an interface by using the `no ipv6 enable` command deletes all IPv6 neighbor discovery cache entries configured for that interface except static entries (the state of the entry changes to INCMP [Incomplete]).

- Static entries in the IPv6 neighbor discovery cache are not modified by the neighbor discovery process.

- The `clear ipv6 neighbor` command does not remove static entries from the IPv6 neighbor discovery cache; it only clears the dynamic entries.
• The ICMP syslogs generated are caused by a regular refresh of IPv6 neighbor entries. The ASA default timer for IPv6 neighbor entry is 30 seconds, so the ASA would generate ICMPv6 neighbor discovery and response packets about every 30 seconds. If the ASA has both failover LAN and state interfaces configured with IPv6 addresses, then every 30 seconds, ICMPv6 neighbor discovery and response packets will be generated by both ASAs for both configured and link-local IPv6 addresses. In addition, each packet will generate several syslogs (ICMP connection and local-host creation or teardown), so it may appear that constant ICMP syslogs are being generated. The refresh time for IPv6 neighbor entry is configurable on the regular data interface, but not configurable on the failover interface. However, the CPU impact for this ICMP neighbor discovery traffic is minimal.

**Default Settings for IPv6 Neighbor Discovery**

Table 26-1 lists the default settings for IPv6 neighbor discovery.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>value for the neighbor solicitation transmission message interval</td>
<td>1000 seconds between neighbor solicitation transmissions.</td>
</tr>
<tr>
<td>value for the neighbor reachable time</td>
<td>The default is 0.</td>
</tr>
<tr>
<td>value for the router advertisement transmission interval</td>
<td>The default is 200 seconds.</td>
</tr>
<tr>
<td>value for the router lifetime</td>
<td>The default is 1800 seconds.</td>
</tr>
<tr>
<td>value for the number of consecutive neighbor solicitation messages sent during DAD</td>
<td>The default is one message.</td>
</tr>
<tr>
<td>prefix lifetime</td>
<td>The default lifetime is 2592000 seconds (30 days), and a preferred lifetime is 604800 seconds (7 days).</td>
</tr>
<tr>
<td>on-link flag</td>
<td>The flag is on by default, which means that the prefix is used on the advertising interface.</td>
</tr>
<tr>
<td>autoconfig flag</td>
<td>The flag is on by default, which means that the prefix is used for autoconfiguration.</td>
</tr>
<tr>
<td>static IPv6 neighbor</td>
<td>Static entries are not configured in the IPv6 neighbor discovery cache.</td>
</tr>
</tbody>
</table>
Configuring the Neighbor Solicitation Message Interval

To configure the interval between IPv6 neighbor solicitation retransmissions on an interface, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>ipv6 nd ns-interval value</td>
<td>Sets the interval between IPv6 neighbor solicitation retransmissions on an interface. Valid values for the value argument range from 1000 to 3600000 milliseconds. This information is also sent in router advertisement messages.</td>
</tr>
</tbody>
</table>

**Example:**

```
hostname (config-if)# ipv6 nd ns-interval 9000
```

**Examples**

The following example configures an IPv6 neighbor solicitation transmission interval of 9000 milliseconds for GigabitEthernet 0/0:

```
hostname (config)# interface gigabitethernet 0/0
hostname (config-if)# ipv6 nd ns-interval 9000
```

Configuring the Neighbor Reachable Time

To configure the amount of time that a remote IPv6 node is considered reachable after a reachability confirmation event has occurred, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>ipv6 nd reachable-time value</td>
<td>Sets the amount of time that a remote IPv6 node is reachable. Valid values for the value argument range from 0 to 3600000 milliseconds. When 0 is used for the value, the reachable time is sent as undetermined. It is up to the receiving devices to set and track the reachable time value.</td>
</tr>
</tbody>
</table>

**Example:**

```
hostname (config-if)# ipv6 nd reachable-time 1700000
```

**Examples**

The following example configures an IPv6 reachable time of 1700000 milliseconds for the selected interface, GigabitEthernet 0/0:

```
hostname (config)# interface gigabitethernet 0/0
hostname (config-if)# ipv6 nd reachable-time 1700000
```
Configuring the Router Advertisement Transmission Interval

To configure the interval between IPv6 router advertisement transmissions on an interface, enter the following command:

```
ipv6 nd ra-interval [msec] value
```

**Example:**
```
hostname (config-if)# ipv6 nd ra-interval 201
```

### Configuring the Router Lifetime Value

To configure the router lifetime value in IPv6 router advertisements on an interface, enter the following command:

```
ipv6 nd ra-lifetime [msec] value
```

**Example:**
```
hostname (config-if)# ipv6 nd ra-lifetime 2000
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>ipv6 nd ra-interval [msec] value</td>
<td>Sets the interval between IPv6 router advertisement transmissions. The optional <code>msec</code> keyword indicates that the value provided is in milliseconds. If this keyword is not present, the value provided is in seconds. Valid values for the <code>value</code> argument range from 3 to 1800 seconds or from 500 to 1800000 milliseconds if the <code>msec</code> keyword is provided. The interval between transmissions should be less than or equal to the IPv6 router advertisement lifetime if the ASASM is configured as a default router. For more information, see the “Configuring the Router Lifetime Value” section on page 26-8. To prevent synchronization with other IPv6 nodes, randomly adjust the actual value used to within 20 percent of the desired value.</td>
</tr>
<tr>
<td>ipv6 nd ra-lifetime [msec] value</td>
<td>Specifies the length of time that nodes on the local link should consider the ASASM as the default router on the link. The optional <code>msec</code> keyword indicates that the value provided is in milliseconds. If this keyword is not present, the value provided is in seconds. Valid values for the <code>value</code> argument range from 0 to 9000 seconds. Entering 0 indicates that the ASASM should not be considered a default router on the selected interface.</td>
</tr>
</tbody>
</table>

Examples

The following example configures an IPv6 router advertisement interval of 201 seconds for the selected interface, GigabitEthernet 0/0:

```
hostname (config)# interface gigabitethernet 0/0
hostname (config-if)# ipv6 nd ra-interval 201
```
Examples

The following example configures an IPv6 router lifetime value of 2000 seconds for the selected interface, GigabitEthernet 0/0:

hostname (config)# interface gigabitethernet 0/0
hostname (config-if)# ipv6 nd ra-lifetime 2000

Configuring DAD Settings

To specify DAD settings on the interface, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>ipv6 nd dad attempts value</td>
<td>Specifies the uniqueness of new unicast IPv6 addresses before they are assigned and ensures that duplicate IPv6 addresses are detected in the network on a link basis.</td>
</tr>
</tbody>
</table>

Example:

hostname (config-if)# ipv6 nd dad attempts 20

Valid values for the value argument range from 0 to 600. A zero value disables DAD processing on the specified interface.

Examples

The following example configures a DAD attempt value of 20 for the selected interface, GigabitEthernet 0/0:

hostname (config)# interface gigabitethernet 0/0
hostname (config-if)# ipv6 nd dad attempts 20

Configuring IPv6 Addresses on an Interface

To configure IPv6 addresses on an interface, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>ipv6 address</td>
<td>Specifies the IPv6 address for the selected interface.</td>
</tr>
</tbody>
</table>

Example:

hostname (config-if)# ipv6 address fe80::20d:88ff:feee:6a82

Examples

The following example configures a link-local IPv6 address for the selected interface, GigabitEthernet 0/0:

hostname (config)# interface gigabitethernet 0/0
hostname (config-if)# ipv6 address fe80::20d:88ff:feee:6a82
Suppressing Router Advertisement Messages

Router advertisement messages are automatically sent in response to router solicitation messages. You may want to disable these messages on any interface for which you do not want the ASASM to supply the IPv6 prefix (for example, the outside interface).

To suppress the router lifetime value in IPv6 router advertisements on an interface, enter the following command:

```
Example:
hostname (config-if)# ipv6 nd suppress-ra 2001:DB8::/32 1000 900
```

### Command

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>ipv6 nd suppress-ra</td>
<td>Suppresses the router lifetime value.</td>
</tr>
<tr>
<td>seconds</td>
<td></td>
</tr>
</tbody>
</table>

Example:
hostname (config-if)# ipv6 nd suppress-ra 2001:DB8::/32 1000 900

**Examples**

The following example suppresses an IPv6 router advertisement transmission for the specified interface, which is GigabitEthernet 0/0:

```
hostname (config)# interface gigabitethernet 0/0
hostname (config-if)# ipv6 nd suppress-ra 2001:DB8::/32 1000 900
```
# Configuring the IPv6 Prefix

To configure the which IPv6 prefixes are included in IPv6 router advertisements, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>ipv6 nd prefix ipv6-prefix/prefix-length</td>
<td>Configures which IPv6 prefixes are included in IPv6 router advertisements. The prefix advertisement can be used by neighboring devices to autoconfigure their interface addresses. Stateless autoconfiguration uses IPv6 prefixes provided in router advertisement messages to create the global unicast address from the link-local address. The at valid-date preferred-date syntax indicates the date and time at which the lifetime and preference expire. The prefix is valid until this specified date and time are reached. Dates are expressed in the form date-valid-expire month-valid-expire hh:mm-valid-expire date-prefer-expire month-prefer-expire hh:mm-prefer-expire. The default keyword indicates that default values are used. The optional infinite keyword specifies that the valid lifetime does not expire. The ipv6-prefix argument specifies the IPv6 network number to include in router advertisements. This argument must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons. The no-advertise keyword indicates to hosts on the local link that the specified prefix is not to be used for IPv6 autoconfiguration. The no-autoconfig keyword indicates to hosts on the local link that the specified prefix cannot be used for IPv6 autoconfiguration. The off-link keyword indicates that the specified prefix is not used for on-link determination. The preferred-lifetime argument specifies the amount of time (in seconds) that the specified IPv6 prefix is advertised as being preferred. Valid values range from 0 to 4294967295 seconds. The maximum value represents infinity, which can also be specified with infinite. The default is 604800 (7 days). The prefix-length argument specifies the length of the IPv6 prefix. This value indicates how many of the high-order, contiguous bits of the address comprise the network portion of the prefix. The slash (/) must precede the prefix length. The valid-lifetime argument specifies the amount of time that the specified IPv6 prefix is advertised as being valid. Valid values range from 0 to 4294967295 seconds. The maximum value represents infinity, which can also be specified with infinite. The default is 2592000 (30 days).</td>
</tr>
</tbody>
</table>

Example:
hostname (config-if)# ipv6 nd prefix 2001:DB8::/32 1000 900
Examples

The following example includes the IPv6 prefix 2001:DB8::/32, with a valid lifetime of 1000 seconds and a preferred lifetime of 900 seconds, in router advertisements sent out on the specified interface, which is GigabitEthernet 0/0:

```bash
hostname (config)# interface gigabitethernet 0/0
hostname (config-if)# ipv6 nd prefix 2001:DB8::/32 1000 900
```

Configuring a Static IPv6 Neighbor

To configure a static entry in the IPv6 neighbor discovery cache, enter the following command:

```
Command Purpose
ipv6 neighbor ipv6_address if_name mac_address

Example:
hostname(config-if)# ipv6 neighbor 3001:1::45A inside 002.7D1A.9472
```

Configures a static entry in the IPv6 neighbor discovery cache. The `ipv6_address` argument is the link-local IPv6 address of the neighbor, the `if_name` argument is the interface through which the neighbor is available, and the `mac_address` argument is the MAC address of the neighbor interface.

Examples

The following example adds a static entry for an inside host with an IPv6 address of 3001:1::45A and a MAC address of 002.7D1A.9472 to the neighbor discovery cache:

```bash
hostname(config-if)# ipv6 neighbor 3001:1::45A inside 002.7D1A.9472
```
Monitoring IPv6 Neighbor Discovery

To monitor IPv6 neighbor discovery parameters, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show ipv6 interface</code></td>
<td>Displays the usability status of interfaces configured for IPv6. Including the interface name, such as “outside” and displays the settings for the specified interface. Excludes the name from the command and displays the settings for all interfaces that have IPv6 enabled on them. Output for the command shows the following:</td>
</tr>
<tr>
<td></td>
<td>• The name and status of the interface.</td>
</tr>
<tr>
<td></td>
<td>• The link-local and global unicast addresses.</td>
</tr>
<tr>
<td></td>
<td>• The multicast groups to which the interface belongs.</td>
</tr>
<tr>
<td></td>
<td>• ICMP redirect and error message settings.</td>
</tr>
<tr>
<td></td>
<td>• Neighbor discovery settings.</td>
</tr>
<tr>
<td></td>
<td>• The actual time when the command is set to 0.</td>
</tr>
<tr>
<td></td>
<td>• The neighbor discovery reachable time that is being used.</td>
</tr>
</tbody>
</table>

Additional References

For additional information related to implementing IPv6 prefixes, see the following topics:

- Related Documents for IPv6 Prefixes, page 26-14
- RFCs for IPv6 Prefixes and Documentation, page 26-14
Related Documents for IPv6 Prefixes

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>ipv6 commands</td>
<td>command reference</td>
</tr>
</tbody>
</table>

RFCs for IPv6 Prefixes and Documentation

<table>
<thead>
<tr>
<th>RFC</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFC 2373</td>
<td>IP Version 6 Addressing Architecture</td>
</tr>
<tr>
<td>RFC 3849</td>
<td>IPv6 Address Prefix Reserved for Documentation</td>
</tr>
</tbody>
</table>

RFC 2373 includes complete documentation to show how IPv6 network address numbers must be shown in router advertisements. The command argument ipv6-prefix indicates this network number, in which the address must be specified in hexadecimal format using 16-bit values between colons.

RFC 3849 specifies the requirements for using IPv6 address prefixes in documentation. The IPv6 unicast address prefix that has been reserved for use in documentation is 2001:DB8::/32.

Feature History for IPv6 Neighbor Discovery

Table 26-2 lists each feature change and the platform release in which it was implemented.

Table 26-2 Feature History for IPv6 Neighbor Discovery

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPv6 Neighbor Discovery</td>
<td>7.0(1)</td>
<td>We introduced this feature.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>We introduced the following commands: ipv6 nd ns-interval, ipv6 nd ra-lifetime, ipv6 nd suppress-ra, ipv6 neighbor, ipv6 nd prefix, ipv6 nd dad-attempts, ipv6 nd reachable-time, ipv6 address, ipv6 enforce-eui64.</td>
</tr>
</tbody>
</table>
P A R T  7

Configuring Network Address Translation
Information About NAT

This chapter provides an overview of how Network Address Translation (NAT) works on the ASASM. This chapter includes the following sections:

- Why Use NAT?, page 27-1
- NAT Terminology, page 27-2
- NAT Types, page 27-3
- NAT in Routed and Transparent Mode, page 27-12
- NAT for VPN, page 27-14
- How NAT is Implemented, page 27-16
- NAT Rule Order, page 27-20
- Routing NAT Packets, page 27-21
- DNS and NAT, page 27-24
- Where to Go Next, page 27-27

Note
To start configuring NAT, see Chapter 28, “Configuring Network Object NAT,” or Chapter 29, “Configuring Twice NAT.”

Why Use NAT?

Each computer and device within an IP network is assigned a unique IP address that identifies the host. Because of a shortage of public IPv4 addresses, most of these IP addresses are private, not routable anywhere outside of the private company network. RFC 1918 defines the private IP addresses you can use internally that should not be advertised:

- 10.0.0.0 through 10.255.255.255
- 172.16.0.0 through 172.31.255.255
- 192.168.0.0 through 192.168.255.255
One of the main functions of NAT is to enable private IP networks to connect to the Internet. NAT replaces a private IP address with a public IP address, translating the private addresses in the internal private network into legal, routable addresses that can be used on the public Internet. In this way, NAT conserves public addresses because it can be configured to advertise at a minimum only one public address for the entire network to the outside world.

Other functions of NAT include:

- Security—Keeping internal IP addresses hidden discourages direct attacks.
- IP routing solutions—Overlapping IP addresses are not a problem when you use NAT.
- Flexibility—You can change internal IP addressing schemes without affecting the public addresses available externally; for example, for a server accessible to the Internet, you can maintain a fixed IP address for Internet use, but internally, you can change the server address.

**Note**

NAT is not required. If you do not configure NAT for a given set of traffic, that traffic will not be translated, but will have all of the security policies applied as normal.

**NAT Terminology**

This document uses the following terminology:

- Real address/host/network/interface—The real address is the address that is defined on the host, before it is translated. In a typical NAT scenario where you want to translate the inside network when it accesses the outside, the inside network would be the “real” network. Note that you can translate any network connected to the ASASM, not just an inside network. Therefore if you configure NAT to translate outside addresses, “real” can refer to the outside network when it accesses the inside network.

- Mapped address/host/network/interface—The mapped address is the address that the real address is translated to. In a typical NAT scenario where you want to translate the inside network when it accesses the outside, the outside network would be the “mapped” network.

- Bidirectional initiation—Static NAT allows connections to be initiated bidirectionally, meaning both to the host and from the host.

- Source and destination NAT—For any given packet, both the source and destination IP addresses are compared to the NAT rules, and one or both can be translated/untranslated. For static NAT, the rule is bidirectional, so be aware that “source” and “destination” are used in commands and descriptions throughout this guide even though a given connection might originate at the “destination” address.
NAT Types

- NAT Types Overview, page 27-3
- Static NAT, page 27-3
- Dynamic NAT, page 27-8
- Dynamic PAT, page 27-10
- Identity NAT, page 27-11

NAT Types Overview

You can implement NAT using the following methods:
- Static NAT—A consistent mapping between a real and mapped IP address. Allows bidirectional traffic initiation. See the “Static NAT” section on page 27-3.
- Dynamic NAT—A group of real IP addresses are mapped to a (usually smaller) group of mapped IP addresses, on a first come, first served basis. Only the real host can initiate traffic. See the “Dynamic NAT” section on page 27-8.
- Dynamic Port Address Translation (PAT)—A group of real IP addresses are mapped to a single IP address using a unique source port of that IP address. See the “Dynamic PAT” section on page 27-10.
- Identity NAT—A real address is statically translated to itself, essentially bypassing NAT. You might want to configure NAT this way when you want to translate a large group of addresses, but then want to exempt a smaller subset of addresses. See the “Identity NAT” section on page 27-11.

Static NAT

This section describes static NAT and includes the following topics:
- Information About Static NAT, page 27-3
- Information About Static NAT with Port Translation, page 27-4
- Information About One-to-Many Static NAT, page 27-6
- Information About Other Mapping Scenarios (Not Recommended), page 27-7

Information About Static NAT

Static NAT creates a fixed translation of a real address to a mapped address. Because the mapped address is the same for each consecutive connection, static NAT allows bidirectional connection initiation, both to and from the host (if an access rule exists that allows it). With dynamic NAT and PAT, on the other hand, each host uses a different address or port for each subsequent translation, so bidirectional initiation is not supported.
Figure 27-1 shows a typical static NAT scenario. The translation is always active so both real and remote hosts can initiate connections.

**Information About Static NAT with Port Translation**

Static NAT with port translation lets you specify a real and mapped protocol (TCP or UDP) and port. This section includes the following topics:

- Information About Static NAT with Port Address Translation, page 27-4
- Static NAT with Identity Port Translation, page 27-5
- Static NAT with Port Translation for Non-Standard Ports, page 27-5
- Static Interface NAT with Port Translation, page 27-5

**Information About Static NAT with Port Address Translation**

When you specify the port with static NAT, you can choose to map the port and/or the IP address to the same value or to a different value.

Figure 27-2 shows a typical static NAT with port translation scenario showing both a port that is mapped to itself and a port that is mapped to a different value; the IP address is mapped to a different value in both cases. The translation is always active so both translated and remote hosts can initiate connections.

**Note**

For applications that require application inspection for secondary channels (for example, FTP and VoIP), the ASASM automatically translates the secondary ports.
Static NAT with Identity Port Translation

The following static NAT with port translation example provides a single address for remote users to access FTP, HTTP, and SMTP. These servers are actually different devices on the real network, but for each server, you can specify static NAT with port translation rules that use the same mapped IP address, but different ports. (See Figure 27-3. See the “Single Address for FTP, HTTP, and SMTP (Static NAT-with-Port-Translation)” section on page 28-18 for details on how to configure this example.)

Figure 27-3  Static NAT with Port Translation

Static NAT with Port Translation for Non-Standard Ports

You can also use static NAT with port translation to translate a well-known port to a non-standard port or vice versa. For example, if inside web servers use port 8080, you can allow outside users to connect to port 80, and then undo translation to the original port 8080. Similarly, to provide extra security, you can tell web users to connect to non-standard port 6785, and then undo translation to port 80.

Static Interface NAT with Port Translation

You can configure static NAT to map a real address to an interface address/port combination. For example, if you want to redirect Telnet access for the ASASM outside interface to an inside host, then you can map the inside host IP address/port 23 to the ASASM interface address/port 23. (Note that although Telnet to the ASASM is not allowed to the lowest security interface, static NAT with interface port translation redirects the Telnet session instead of denying it.)
Information About One-to-Many Static NAT

Typically, you configure static NAT with a one-to-one mapping. However, in some cases, you might want to configure a single real address to several mapped addresses (one-to-many). When you configure one-to-many static NAT, when the real host initiates traffic, it always uses the first mapped address. However, for traffic initiated to the host, you can initiate traffic to any of the mapped addresses, and they will be untranslated to the single real address.

Figure 27-4 shows a typical one-to-many static NAT scenario. Because initiation by the real host always uses the first mapped address, the translation of real host IP/1st mapped IP is technically the only bidirectional translation.

![Figure 27-4 One-to-Many Static NAT](image-url)
For example, you have a load balancer at 10.1.2.27. Depending on the URL requested, it redirects traffic to the correct web server (see Figure 27-5). (See the “Inside Load Balancer with Multiple Mapped Addresses (Static NAT, One-to-Many)” section on page 28-17 for details on how to configure this example.)

**Figure 27-5 One-to-Many Static NAT**

![Diagram showing one-to-many static NAT configuration](image)

### Information About Other Mapping Scenarios (Not Recommended)

The ASASM has the flexibility to allow any kind of static mapping scenario: one-to-one, one-to-many, but also few-to-many, many-to-few, and many-to-one mappings. We recommend using only one-to-one or one-to-many mappings. These other mapping options might result in unintended consequences.

Functionally, few-to-many is the same as one-to-many; but because the configuration is more complicated and the actual mappings may not be obvious at a glance, we recommend creating a one-to-many configuration for each real address that requires it. For example, for a few-to-many scenario, the few real addresses are mapped to the many mapped addresses in order (A to 1, B to 2, C to 3). When all real addresses are mapped, the next mapped address is mapped to the first real address, and so on until all mapped addresses are mapped (A to 4, B to 5, C to 6). This results in multiple mapped addresses for each real address. Just like a one-to-many configuration, only the first mappings are bidirectional; subsequent mappings allow traffic to be initiated to the real host, but all traffic from the real host uses only the first mapped address for the source.
Figure 27-6 shows a typical few-to-many static NAT scenario.

![Figure 27-6](image)

For a many-to-few or many-to-one configuration, where you have more real addresses than mapped addresses, you run out of mapped addresses before you run out of real addresses. Only the mappings between the lowest real IP addresses and the mapped pool result in bidirectional initiation. The remaining higher real addresses can initiate traffic, but traffic cannot be initiated to them (returning traffic for a connection is directed to the correct real address because of the unique 5-tuple (source IP, destination IP, source port, destination port, protocol) for the connection).

**Note**

Many-to-few or many-to-one NAT is not PAT. If two real hosts use the same source port number and go to the same outside server and the same TCP destination port, and both hosts are translated to the same IP address, then both connections will be reset because of an address conflict (the 5-tuple is not unique).

Figure 27-7 shows a typical many-to-few static NAT scenario.

![Figure 27-7](image)

Instead of using a static rule this way, we suggest that you create a one-to-one rule for the traffic that needs bidirectional initiation, and then create a dynamic rule for the rest of your addresses.

**Dynamic NAT**

This section describes dynamic NAT and includes the following topics:

- Information About Dynamic NAT, page 27-9
- Dynamic NAT Disadvantages and Advantages, page 27-10
Information About Dynamic NAT

Dynamic NAT translates a group of real addresses to a pool of mapped addresses that are routable on the destination network. The mapped pool typically includes fewer addresses than the real group. When a host you want to translate accesses the destination network, the ASASM assigns the host an IP address from the mapped pool. The translation is created only when the real host initiates the connection. The translation is in place only for the duration of the connection, and a given user does not keep the same IP address after the translation times out. Users on the destination network, therefore, cannot initiate a reliable connection to a host that uses dynamic NAT, even if the connection is allowed by an access rule.

**Figure 27-8** shows a typical dynamic NAT scenario. Only real hosts can create a NAT session, and responding traffic is allowed back.

**Figure 27-8  Dynamic NAT**

<table>
<thead>
<tr>
<th>Inside</th>
<th>Outside</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.1.1.1</td>
<td>209.165.201.1</td>
</tr>
<tr>
<td>10.1.1.2</td>
<td>209.165.201.2</td>
</tr>
</tbody>
</table>

**Figure 27-9** shows a remote host attempting to initiate a connection to a mapped address. This address is not currently in the translation table; therefore, the ASASM drops the packet.

**Figure 27-9  Remote Host Attempts to Initiate a Connection to a Mapped Address**
Note

For the duration of the translation, a remote host can initiate a connection to the translated host if an access rule allows it. Because the address is unpredictable, a connection to the host is unlikely. Nevertheless, in this case you can rely on the security of the access rule.

Dynamic NAT Disadvantages and Advantages

Dynamic NAT has these disadvantages:

- If the mapped pool has fewer addresses than the real group, you could run out of addresses if the amount of traffic is more than expected.
  Use PAT or a PAT fallback method if this event occurs often because PAT provides over 64,000 translations using ports of a single address.
- You have to use a large number of routable addresses in the mapped pool, and routable addresses may not be available in large quantities.

The advantage of dynamic NAT is that some protocols cannot use PAT. PAT does not work with the following:

- IP protocols that do not have a port to overload, such as GRE version 0.
- Some multimedia applications that have a data stream on one port, the control path on another port, and are not open standard.

See the “Default Settings” section on page 39-4 for more information about NAT and PAT support.

Dynamic PAT

This section describes dynamic PAT and includes the following topics:

- Information About Dynamic PAT, page 27-10
- Dynamic PAT Disadvantages and Advantages, page 27-11

Information About Dynamic PAT

Dynamic PAT translates multiple real addresses to a single mapped IP address by translating the real address and source port to the mapped address and a unique port. If available, the real source port number is used for the mapped port. However, if the real port is not available, by default the mapped ports are chosen from the same range of ports as the real port number: 0 to 511, 512 to 1023, and 1024 to 65535. Therefore, ports below 1024 have only a small PAT pool that can be used.

Each connection requires a separate translation session because the source port differs for each connection. For example, 10.1.1.1:1025 requires a separate translation from 10.1.1.1:1026.
Figure 27-10 shows a typical dynamic PAT scenario. Only real hosts can create a NAT session, and responding traffic is allowed back. The mapped address is the same for each translation, but the port is dynamically assigned.

**Figure 27-10  Dynamic PAT**

![Dynamic PAT Diagram]

After the connection expires, the port translation also expires after 30 seconds of inactivity. The timeout is not configurable. Users on the destination network cannot reliably initiate a connection to a host that uses PAT (even if the connection is allowed by an access rule).

**Note**

For the duration of the translation, a remote host can initiate a connection to the translated host if an access rule allows it. Because the port address (both real and mapped) is unpredictable, a connection to the host is unlikely. Nevertheless, in this case you can rely on the security of the access rule.

### Dynamic PAT Disadvantages and Advantages

Dynamic PAT lets you use a single mapped address, thus conserving routable addresses. You can even use the ASASM interface IP address as the PAT address.

Dynamic PAT does not work with some multimedia applications that have a data stream that is different from the control path. See the “Default Settings” section on page 39-4 for more information about NAT and PAT support.

Dynamic PAT may also create a large number of connections appearing to come from a single IP address, and servers might interpret the traffic as a DoS attack. (8.4(2)/8.5(1) and later) You can configure a PAT pool of addresses and use a round-robin assignment of PAT addresses to mitigate this situation.

### Identity NAT

You might have a NAT configuration in which you need to translate an IP address to itself. For example, if you create a broad rule that applies NAT to every network, but want to exclude one network from NAT, you can create a static NAT rule to translate an address to itself. Identity NAT is necessary for remote access VPN, where you need to exempt the client traffic from NAT.
Figure 27-11 shows a typical identity NAT scenario.

![Identity NAT Diagram]

**NAT in Routed and Transparent Mode**

You can configure NAT in both routed and transparent firewall mode. This section describes typical usage for each firewall mode and includes the following topics:

- NAT in Routed Mode, page 27-13
- NAT in Transparent Mode, page 27-13
**NAT in Routed Mode**

Figure 27-12 shows a typical NAT example in routed mode, with a private network on the inside.

![NAT Example: Routed Mode](image)

1. When the inside host at 10.1.2.27 sends a packet to a web server, the real source address of the packet, 10.1.2.27, is changed to a mapped address, 209.165.201.10.

2. When the server responds, it sends the response to the mapped address, 209.165.201.10, and the ASASM receives the packet because the ASASM performs proxy ARP to claim the packet.

3. The ASASM then changes the translation of the mapped address, 209.165.201.10, back to the real address, 10.1.2.27, before sending it to the host.

**NAT in Transparent Mode**

Using NAT in transparent mode eliminates the need for the upstream or downstream routers to perform NAT for their networks.

NAT in transparent mode has the following requirements and limitations:

- Because the transparent firewall does not have any interface IP addresses, you cannot use interface PAT.

- ARP inspection is not supported. Moreover, if for some reason a host on one side of the ASASM sends an ARP request to a host on the other side of the ASASM, and the initiating host real address is mapped to a different address on the same subnet, then the real address remains visible in the ARP request.

Figure 27-13 shows a typical NAT scenario in transparent mode, with the same network on the inside and outside interfaces. The transparent firewall in this scenario is performing the NAT service so that the upstream router does not have to perform NAT.
1. When the inside host at 10.1.1.75 sends a packet to a web server, the real source address of the packet, 10.1.1.75, is changed to a mapped address, 209.165.201.15.

2. When the server responds, it sends the response to the mapped address, 209.165.201.15, and the ASASM receives the packet because the upstream router includes this mapped network in a static route directed to the ASASM management IP address. See the “Mapped Addresses and Routing” section on page 27-22 for more information about required routes.

3. The ASASM then undoes the translation of the mapped address, 209.165.201.15, back to the real address, 10.1.1.75. Because the real address is directly-connected, the ASASM sends it directly to the host.

4. For host 192.168.1.2, the same process occurs, except for returning traffic, the ASASM looks up the route in its routing table and sends the packet to the downstream router at 10.1.1.3 based on the ASASM static route for 192.168.1.0/24. See the “Transparent Mode Routing Requirements for Remote Networks” section on page 27-23 for more information about required routes.

NAT for VPN

If you do not allow split-tunneling, then all VPN traffic, even traffic destined for the Internet, goes through the VPN tunnel. VPN traffic, after being decrypted by the ASASM, is essentially the same as any other inside traffic: when an inside user needs to access the Internet, they need a public IP address provided by NAT.
Figure 27-14 shows a VPN client that wants to visit a website at www.example.com. In this example, an interface PAT rule on the outside interface matches the VPN-assigned address 10.1.1.10. With intra-interface communication enabled, traffic can exit the same interface it entered to reach www.example.com. A similar example without the need for hairpin networking includes an ASASM for VPN termination, and a separate ASASM with NAT as the Internet gateway.

**Figure 27-14  Interface PAT for Internet-Bound VPN Traffic (Hairpin, Intra-Interface)**

1. HTTP request to www.example.com
2. ASA replaces src address with local address

```
209.165.201.10 ➔ 10.1.1.10
```

3. ASA performs interface PAT for outgoing traffic. 

**Note:** This “hairpin” traffic flow requires you to enable intra-interface communication.

**Figure 27-15  Identity NAT to Allow Communication Between VPN Sites and Clients**

1. IM to 10.2.2.78
2. Identity NAT for 10.1.1.0, 10.2.2.0, & 10.3.3.0 networks when going to other inside networks connected by VPN

```
10.1.1.6 ➔ 10.1.1.16
```

3. IM received
4. HTTP request to www.example.com

**Figure 27-15** also shows an interface PAT rule for Internet-bound traffic. However, for any communication between VPN endpoints such as the ends of a site-to-site tunnel, you do not want to perform NAT. Therefore you also need to create an identity NAT rule (using twice NAT) for any traffic that goes to other inside networks connected by VPN.
How NAT is Implemented

The ASASM can implement address translation in two ways: network object NAT and twice NAT. This section includes the following topics:

- Main Differences Between Network Object NAT and Twice NAT, page 27-16
- Information About Network Object NAT, page 27-17
- Information About Twice NAT, page 27-17

Main Differences Between Network Object NAT and Twice NAT

The main differences between these two NAT types are:

- How you define the real address.
  - Network object NAT—You define NAT as a parameter for a network object. A network object names an IP host, range, or subnet so you can then use the object in configuration instead of the actual IP addresses. The network object IP address serves as the real address. This method lets you easily add NAT to network objects that might already be used in other parts of your configuration.
  - Twice NAT—You identify a network object or network object group for both the real and mapped addresses. In this case, NAT is not a parameter of the network object; the network object or group is a parameter of the NAT configuration. The ability to use a network object group for the real address means that twice NAT is more scalable.

- How source and destination NAT is implemented.
  - Network object NAT—Each rule can apply to either the source or destination of a packet. So two rules might be used, one for the source IP address, and one for the destination IP address. These two rules cannot be tied together to enforce a specific translation for a source/destination combination.
  - Twice NAT—A single rule translates both the source and destination. A matching packet only matches the one rule, and further rules are not checked. Even if you do not configure the optional destination address for twice NAT, a matching packet still only matches one twice NAT rule. The source and destination are tied together, so you can enforce different translations depending on the source/destination combination. For example, sourceA/destinationA can have a different translation than sourceA/destinationB.

- Order of NAT Rules.
  - Network object NAT—Automatically ordered in the NAT table.
  - Twice NAT—Manually ordered in the NAT table (before or after network object NAT rules).

See the “NAT Rule Order” section on page 27-20 for more information.

We recommend using network object NAT unless you need the extra features that twice NAT provides. Network object NAT is easier to configure, and might be more reliable for applications such as Voice over IP (VoIP). (For VoIP, because twice NAT is applicable only between two objects, you might see a failure in the translation of indirect addresses that do not belong to either of the objects.)
Information About Network Object NAT

All NAT rules that are configured as a parameter of a network object are considered to be network object NAT rules. Network object NAT is a quick and easy way to configure NAT for a network object, which can be a single IP address, a range of addresses, or a subnet.

After you configure the network object, you can then identify the mapped address for that object, either as an inline address or as another network object or network object group.

When a packet enters the ASASM, both the source and destination IP addresses are checked against the network object NAT rules. The source and destination address in the packet can be translated by separate rules if separate matches are made. These rules are not tied to each other; different combinations of rules can be used depending on the traffic.

Because the rules are never paired, you cannot specify that sourceA/destinationA should have a different translation than sourceA/destinationB. Use twice NAT for that kind of functionality (twice NAT lets you identify the source and destination address in a single rule).

To start configuring network object NAT, see Chapter 28, “Configuring Network Object NAT.”

Information About Twice NAT

Twice NAT lets you identify both the source and destination address in a single rule. Specifying both the source and destination addresses lets you specify that sourceA/destinationA can have a different translation than sourceA/destinationB.

The destination address is optional. If you specify the destination address, you can either map it to itself (identity NAT), or you can map it to a different address. The destination mapping is always a static mapping.

Twice NAT also lets you use service objects for static NAT with port translation; network object NAT only accepts inline definition.

To start configuring twice NAT, see Chapter 29, “Configuring Twice NAT.”

Figure 27-16 shows a host on the 10.1.2.0/24 network accessing two different servers. When the host accesses the server at 209.165.201.11, the real address is translated to 209.165.202.129. When the host accesses the server at 209.165.200.225, the real address is translated to 209.165.202.130. (See the “Single Address for FTP, HTTP, and SMTP (Static NAT-with-Port-Translation)” section on page 28-18 for details on how to configure this example.)
Figure 27-16  Twice NAT with Different Destination Addresses

Figure 27-17 shows the use of source and destination ports. The host on the 10.1.2.0/24 network accesses a single host for both web services and Telnet services. When the host accesses the server for web services, the real address is translated to 209.165.202.129. When the host accesses the same server for Telnet services, the real address is translated to 209.165.202.130.
Figure 27-18 shows a remote host connecting to a mapped host. The mapped host has a twice static NAT translation that translates the real address only for traffic to and from the 209.165.201.0/27 network. A translation does not exist for the 209.165.200.224/27 network, so the translated host cannot connect to that network, nor can a host on that network connect to the translated host.

Figure 27-18  Twice Static NAT with Destination Address Translation

![Twice Static NAT with Destination Address Translation Diagram](image-url)
NAT Rule Order

Network object NAT rules and twice NAT rules are stored in a single table that is divided into three sections. Section 1 rules are applied first, then section 2, and finally section 3. Table 27-1 shows the order of rules within each section.

Table 27-1  NAT Rule Table

<table>
<thead>
<tr>
<th>Table Section</th>
<th>Rule Type</th>
<th>Order of Rules within the Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>Section 1</td>
<td>Twice NAT</td>
<td>Applied on a first match basis, in the order they appear in the configuration. By default, twice NAT rules are added to section 1.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note</strong> If you configure EasyVPN remote, the ASASM dynamically adds invisible NAT rules to the end of this section. Be sure that you do not configure a twice NAT rule in this section that might match your VPN traffic, instead of matching the invisible rule. If VPN does not work due to NAT failure, consider adding twice NAT rules to section 3 instead.</td>
</tr>
<tr>
<td>Section 2</td>
<td>Network object NAT</td>
<td>Section 2 rules are applied in the following order, as automatically determined by the ASASM:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1. Static rules.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2. Dynamic rules.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Within each rule type, the following ordering guidelines are used:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>a. Quantity of real IP addresses—From smallest to largest. For example, an object with one address will be assessed before an object with 10 addresses.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>b. For quantities that are the same, then the IP address number is used, from lowest to highest. For example, 10.1.1.0 is assessed before 11.1.1.0.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>c. If the same IP address is used, then the name of the network object is used, in alphabetical order. For example, abracadabra is assessed before catwoman.</td>
</tr>
<tr>
<td>Section 3</td>
<td>Twice NAT</td>
<td>Section 3 rules are applied on a first match basis, in the order they appear in the configuration. You can specify whether to add a twice NAT rule to section 3 when you add the rule.</td>
</tr>
</tbody>
</table>

For section 2 rules, for example, you have the following IP addresses defined within network objects:

- 192.168.1.0/24 (static)
- 192.168.1.0/24 (dynamic)
- 10.1.1.0/24 (static)
- 192.168.1.1/32 (static)
- 172.16.1.0/24 (dynamic) (object def)
- 172.16.1.0/24 (dynamic) (object abc)
The resultant ordering would be:

- 192.168.1.1/32 (static)
- 10.1.1.0/24 (static)
- 192.168.1.0/24 (static)
- 172.16.1.0/24 (dynamic) (object abc)
- 172.16.1.0/24 (dynamic) (object def)
- 192.168.1.0/24 (dynamic)

**NAT Interfaces**

You can configure a NAT rule to apply to any interface (in other words, all interfaces), or you can identify specific real and mapped interfaces. You can also specify any interface for the real address, and a specific interface for the mapped address, or vice versa.

For example, you might want to specify any interface for the real address and specify the outside interface for the mapped address if you use the same private addresses on multiple interfaces, and you want to translate them all to the same global pool when accessing the outside (Figure 27-19).

![Figure 27-19 Specifying Any Interface](image)

**Note**

For transparent mode, you must choose specific source and destination interfaces.

**Routing NAT Packets**

The ASASM needs to be the destination for any packets sent to the mapped address. The ASA also needs to determine the egress interface for translated packets. This section describes how the ASASM handles accepting and delivering packets with NAT, and includes the following topics:

- Mapped Addresses and Routing, page 27-22
- Transparent Mode Routing Requirements for Remote Networks, page 27-23
- Determining the Egress Interface, page 27-24
Mapped Addresses and Routing

When you translate the real address to a mapped address, the mapped address you choose determines how to configure routing, if necessary, for the mapped address.

See additional guidelines about mapped IP addresses in Chapter 28, “Configuring Network Object NAT,” and Chapter 29, “Configuring Twice NAT.”

See the following mapped address types:

- **Addresses on the same network as the mapped interface.**
  If you use addresses on the same network as the mapped interface, the ASASM uses proxy ARP to answer any ARP requests for the mapped addresses, thus intercepting traffic destined for a mapped address. This solution simplifies routing because the ASASM does not have to be the gateway for any additional networks. This solution is ideal if the outside network contains an adequate number of free addresses, a consideration if you are using a 1:1 translation like dynamic NAT or static NAT. Dynamic PAT greatly extends the number of translations you can use with a small number of addresses, so even if the available addresses on the outside network is small, this method can be used. For PAT, you can even use the IP address of the mapped interface.

  **Note** If you configure the mapped interface to be any interface, and you specify a mapped address on the same network as one of the mapped interfaces, then if an ARP request for that mapped address comes in on a different interface, then you need to manually configure an ARP entry for that network on the ingress interface, specifying its MAC address (see the `arp` command). Typically, if you specify any interface for the mapped interface, then you use a unique network for the mapped addresses, so this situation would not occur.

- **Addresses on a unique network.**
  If you need more addresses than are available on the mapped interface network, you can identify addresses on a different subnet. The upstream router needs a static route for the mapped addresses that points to the ASASM. Alternatively for routed mode, you can configure a static route on the ASASM for the mapped addresses, and then redistribute the route using your routing protocol. For transparent mode, if the real host is directly-connected, configure the static route on the upstream router to point to the ASASM: in 8.3, specify the global management IP address; in 8.4(1) and later, specify the bridge group IP address. For remote hosts in transparent mode, in the static route on the upstream router, you can alternatively specify the downstream router IP address.

- **The same address as the real address (identity NAT).**
  The default behavior for identity NAT has proxy ARP enabled, matching other static NAT rules. You can disable proxy ARP if desired. **Note:** You can also disable proxy ARP for regular static NAT if desired, in which case you need to be sure to have proper routes on the upstream router.

  Normally for identity NAT, proxy ARP is not required, and in some cases can cause connectivity issues. For example, if you configure a broad identity NAT rule for “any” IP address, then leaving proxy ARP enabled can cause problems for hosts on the network directly-connected to the mapped interface. In this case, when a host on the mapped network wants to communicate with another host on the same network, then the address in the ARP request matches the NAT rule (which matches “any” address). The ASASM will then proxy ARP for the address, even though the packet is not actually destined for the ASASM. (Note that this problem occurs even if you have a twice NAT rule; although the NAT rule must match both the source and destination addresses, the proxy ARP decision is made only on the “source” address). If the ASASM ARP response is received before the actual host ARP response, then traffic will be mistakenly sent to the ASASM (see Figure 27-20).
In rare cases, you need proxy ARP for identity NAT; for example for virtual Telnet. When using AAA for network access, a host needs to authenticate with the ASASM using a service like Telnet before any other traffic can pass. You can configure a virtual Telnet server on the ASASM to provide the necessary login. When accessing the virtual Telnet address from the outside, you must configure an identity NAT rule for the address specifically for the proxy ARP functionality. Due to internal processes for virtual Telnet, proxy ARP lets the ASASM keep traffic destined for the virtual Telnet address rather than send the traffic out the source interface according to the NAT rule. (See Figure 27-21).

**Figure 27-21  Proxy ARP and Virtual Telnet**

**Transparent Mode Routing Requirements for Remote Networks**

If the ASASM performs NAT for a host that is not on the directly-connected network, then you need to configure a static route on the ASASM for that network. You also need to have a static route for embedded IP addresses that are at least one hop away from the ASASM (such as in VoIP or DNS traffic) when you have inspection and NAT enabled.
Determining the Egress Interface

In transparent mode, the ASASM determines the egress interface for a NAT packet by using the NAT configuration; you must specify the source and destination interfaces as part of the NAT configuration.

In routed mode, the ASASM determines the egress interface for a NAT packet in the following way:

- If you specify an optional interface, then the ASASM uses the NAT configuration to determine the egress interface. For identity NAT, the default behavior is to use the NAT configuration, but you have the option to always use a route lookup instead.
- If you do not specify a specific interface, then the ASASM uses a route lookup to determine the egress interface.

DNS and NAT

You might need to configure the ASASM to modify DNS replies by replacing the address in the reply with an address that matches the NAT configuration. You can configure DNS modification when you configure each translation rule.

This feature rewrites the A record, or address record, in DNS replies that match a NAT rule. For DNS replies traversing from a mapped interface to any other interface, the A record is rewritten from the mapped value to the real value. Inversely, for DNS replies traversing from any interface to a mapped interface, the A record is rewritten from the real value to the mapped value.

Note

If you configure a twice NAT rule, you cannot configure DNS modification if you specify the source address as well as the destination address. These kinds of rules can potentially have a different translation for a single address when going to A vs. B. Therefore, the ASASM cannot accurately match the IP address inside the DNS reply to the correct twice NAT rule; the DNS reply does not contain information about which source/destination address combination was in the packet that prompted the DNS request.

Figure 27-22 shows a DNS server that is accessible from the outside interface. A server, ftp.cisco.com, is on the inside interface. You configure the ASASM to statically translate the ftp.cisco.com real address (10.1.3.14) to a mapped address (209.165.201.10) that is visible on the outside network. In this case, you want to enable DNS reply modification on this static rule so that inside users who have access to ftp.cisco.com using the real address receive the real address from the DNS server, and not the mapped address. When an inside host sends a DNS request for the address of ftp.cisco.com, the DNS server replies with the mapped address (209.165.201.10). The ASASM refers to the static rule for the inside
server and translates the address inside the DNS reply to 10.1.3.14. If you do not enable DNS reply modification, then the inside host attempts to send traffic to 209.165.201.10 instead of accessing ftp.cisco.com directly.

**Figure 27-22 DNS Reply Modification, DNS Server on Outside**

Figure 27-23 shows a user on the inside network requesting the IP address for ftp.cisco.com, which is on the DMZ network, from an outside DNS server. The DNS server replies with the mapped address (209.165.201.10) according to the static rule between outside and DMZ even though the user is not on the DMZ network. The ASASM translates the address inside the DNS reply to 10.1.3.14. If the user needs to access ftp.cisco.com using the real address, then no further configuration is required. If there is
also a static rule between the inside and DMZ, then you also need to enable DNS reply modification on this rule. The DNS reply will then be modified two times. In this case, the ASASM again translates the address inside the DNS reply to 192.168.1.10 according to the static rule between inside and DMZ.

_Figure 27-23  DNS Reply Modification, DNS Server, Host, and Server on Separate Networks_
Figure 27-24 shows a web server and DNS server on the outside. The ASASM has a static translation for the outside server. In this case, when an inside user requests the address for ftp.cisco.com from the DNS server, the DNS server responds with the real address, 209.165.20.10. Because you want inside users to use the mapped address for ftp.cisco.com (10.1.2.56) you need to configure DNS reply modification for the static translation.

**Figure 27-24   DNS Reply Modification, DNS Server on Host Network**

![Diagram](image)

Where to Go Next

To configure network object NAT, see Chapter 28, “Configuring Network Object NAT.”
To configure twice NAT, see Chapter 29, “Configuring Twice NAT.”
Configuring Network Object NAT

All NAT rules that are configured as a parameter of a network object are considered to be network object NAT rules. Network object NAT is a quick and easy way to configure NAT for a single IP address, a range of addresses, or a subnet. After you configure the network object, you can then identify the mapped address for that object.

This chapter describes how to configure network object NAT, and it includes the following sections:

- Information About Network Object NAT, page 28-1
- Licensing Requirements for Network Object NAT, page 28-2
- Prerequisites for Network Object NAT, page 28-2
- Guidelines and Limitations, page 28-2
- Default Settings, page 28-3
- Configuring Network Object NAT, page 28-3
- Monitoring Network Object NAT, page 28-14
- Configuration Examples for Network Object NAT, page 28-15
- Feature History for Network Object NAT, page 28-22

For detailed information about how NAT works, see Chapter 27, “Information About NAT.”

Information About Network Object NAT

When a packet enters the ASASM, both the source and destination IP addresses are checked against the network object NAT rules. The source and destination address in the packet can be translated by separate rules if separate matches are made. These rules are not tied to each other; different combinations of rules can be used depending on the traffic.

Because the rules are never paired, you cannot specify that a source address should be translated to A when going to destination X, but be translated to B when going to destination Y. Use twice NAT for that kind of functionality (twice NAT lets you identify the source and destination address in a single rule).

For detailed information about the differences between twice NAT and network object NAT, see the “How NAT is Implemented” section on page 27-16.

Network object NAT rules are added to section 2 of the NAT rules table. For more information about NAT ordering, see the “NAT Rule Order” section on page 27-20.
Licensing Requirements for Network Object NAT

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

Prerequisites for Network Object NAT

Depending on the configuration, you can configure the mapped address inline if desired or you can create a separate network object or network object group for the mapped address (the `object network` or `object-group network` command). Network object groups are particularly useful for creating a mapped address pool with discontinuous IP address ranges or multiple hosts or subnets. To create a network object or group, see the “Configuring Objects and Groups” section on page 12-1.

For specific guidelines for objects and groups, see the configuration section for the NAT type you want to configure. See also the “Guidelines and Limitations” section.

Guidelines and Limitations

**Context Mode Guidelines**

Supported in single and multiple context mode.

**Firewall Mode Guidelines**

- Supported in routed and transparent firewall mode.
- In transparent mode, you must specify the real and mapped interfaces; you cannot use `any`.
- In transparent mode, you cannot configure interface PAT, because the transparent mode interfaces do not have IP addresses. You also cannot use the management IP address as a mapped address.

**IPv6 Guidelines**

Does not support IPv6.

**Additional Guidelines**

- You can only define a single NAT rule for a given object; if you want to configure multiple NAT rules for an object, you need to create multiple objects with different names that specify the same IP address, for example, `object network obj-10.10.10.01`, `object network obj-10.10.10.02`, and so on.
Chapter 28      Configuring Network Object NAT

Default Settings

• If you change the NAT configuration, and you do not want to wait for existing translations to time out before the new NAT configuration is used, you can clear the translation table using the `clear xlate` command. However, clearing the translation table disconnects all current connections that use translations.

  **Note**  If you remove a dynamic NAT or PAT rule, and then add a new rule with mapped addresses that overlap the addresses in the removed rule, then the new rule will not be used until all connections associated with the removed rule time out or are cleared using the `clear xlate` command. This safeguard ensures that the same address is not assigned to multiple hosts.

• Objects and object groups used in NAT cannot be undefined; they must include IP addresses.
• You can use the same mapped object or group in multiple NAT rules.
• The mapped IP address pool cannot include:
  – The mapped interface IP address. If you specify `any` interface for the rule, then all interface IP addresses are disallowed. For interface PAT (routed mode only), use the `interface` keyword instead of the IP address.
  – (Transparent mode) The management IP address.
  – (Dynamic NAT) The standby interface IP address when VPN is enabled.
  – Existing VPN pool addresses.
• For application inspection limitations with NAT or PAT, see the “Default Settings” section on page 39-4 in Chapter 39, “Getting Started with Application Layer Protocol Inspection.”

Default Settings

• (Routed mode) The default real and mapped interface is Any, which applies the rule to all interfaces.
• The default behavior for identity NAT has proxy ARP enabled, matching other static NAT rules. You can disable proxy ARP if desired. See the “Routing NAT Packets” section on page 27-21 for more information.
• If you specify an optional interface, then the ASASM uses the NAT configuration to determine the egress interface. For identity NAT, the default behavior is to use the NAT configuration, but you have the option to always use a route lookup instead. See the “Routing NAT Packets” section on page 27-21 for more information.

Configuring Network Object NAT

This section describes how to configure network object NAT and includes the following topics:
• Configuring Dynamic NAT, page 28-4
• Configuring Dynamic PAT (Hide), page 28-6
• Configuring Static NAT or Static NAT-with-Port-Translation, page 28-9
• Configuring Identity NAT, page 28-12
# Configuring Dynamic NAT

This section describes how to configure network object NAT for dynamic NAT. For more information, see the “Dynamic NAT” section on page 27-8.

## Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td>Network object:</td>
<td>To specify the <strong>mapped</strong> addresses (that you want to translate to), configure a network object or network object group. A network object group can contain objects and/or inline addresses.</td>
</tr>
<tr>
<td><code>object network obj_name range ip_address_1 ip_address_2</code></td>
<td></td>
</tr>
<tr>
<td>Network object group:</td>
<td>If a mapped network object contains both ranges and host IP addresses, then the ranges are used for dynamic NAT, and then the host IP addresses are used as a PAT fallback.</td>
</tr>
<tr>
<td><code>object-group network grp_name</code></td>
<td>See the “Guidelines and Limitations” section on page 28-2 for information about disallowed mapped IP addresses.</td>
</tr>
<tr>
<td>`{network-object {object net_obj_name</td>
<td>host ip_address}</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# object network TEST hostname(config-network-object)# range 10.1.1.1 10.1.1.70</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# object network TEST2 hostname(config-network-object)# range 10.1.2.1 10.1.2.70</td>
<td></td>
</tr>
<tr>
<td>hostname(config-network-object)# object-group network MAPPED_IPS hostname(config-network)# network-object object TEST hostname(config-network)# network-object object TEST2 hostname(config-network)# network-object host 10.1.2.79</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Configures a network object for which you want to configure NAT, or enters object network configuration mode for an existing network object.</td>
</tr>
<tr>
<td><code>object network obj_name</code></td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# object network my-host-obj1</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>If you are creating a new network object, defines the <strong>real</strong> IP address(es) that you want to translate.</td>
</tr>
<tr>
<td>`{host ip_address</td>
<td>subnet subnet_address netmask</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config-network-object)# subnet 10.1.1.0 255.255.255.0</td>
<td></td>
</tr>
</tbody>
</table>
Chapter 28  Configuring Network Object NAT

Configuring Network Object NAT

### Examples

The following example configures dynamic NAT that hides 192.168.2.0 network behind a range of outside addresses 10.2.2.1 through 10.2.2.10:

```bash
hostname(config)# object network my-range-obj
hostname(config)# range 10.2.2.1 10.2.2.10
hostname(config)# object network my-inside-net
hostname(config)# subnet 192.168.2.0 255.255.255.0
hostname(config)# nat (inside,outside) dynamic my-range-obj
```

The following example configures dynamic NAT with dynamic PAT backup. Hosts on inside network 10.76.11.0 are mapped first to the nat-range1 pool (10.10.10.10-10.10.10.20). After all addresses in the nat-range1 pool are allocated, dynamic PAT is performed using the pat-ip1 address (10.10.10.21). In the unlikely event that the PAT translations are also use up, dynamic PAT is performed using the outside interface address.

```bash
hostname(config)# object network nat-range1
hostname(config)# range 10.10.10.10 10.10.10.20
hostname(config)# object network pat-ip1
hostname(config)# host 10.10.10.21
hostname(config)# object-group network nat-pat-grp
hostname(config)# network-object object nat-range1
hostname(config)# network-object object pat-ip1
hostname(config)# object network my_net_obj5
hostname(config)# subnet 10.76.11.0 255.255.255.0
hostname(config)# nat (inside,outside) dynamic nat-pat-grp interface
```

### Step 4

```
nat [(real_ifc,mapped_ifc)] dynamic mapped_obj [interface] [dns]
```

**Purpose**

Configures dynamic NAT for the object IP addresses.

**Note**

You can only define a single NAT rule for a given object. See the “Additional Guidelines” section on page 28-2.

See the following guidelines:

- **Interfaces**—(Required for transparent mode) Specify the real and mapped interfaces. Be sure to include the parentheses in your command. In routed mode, if you do not specify the real and mapped interfaces, all interfaces are used; you can also specify the keyword any for one or both of the interfaces.

- **Mapped IP address**—Specify the mapped IP address as:
  - An existing network object (see Step 1).
  - An existing network object group (see Step 1).

- **Interface PAT fallback**—(Optional) The interface keyword enables interface PAT fallback. After the mapped IP addresses are used up, then the IP address of the mapped interface is used. For this option, you must configure a specific interface for the mapped_ifc. (You cannot specify interface in transparent mode).

- **DNS**—(Optional) The dns keyword translates DNS replies. Be sure DNS inspection is enabled (it is enabled by default). See the “DNS and NAT” section on page 27-24 for more information.

### Command Purpose

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>nat [(real_ifc,mapped_ifc)] dynamic mapped_obj [interface] [dns]</td>
<td>Configures dynamic NAT for the object IP addresses.</td>
</tr>
</tbody>
</table>

Example:

```
hostname(config-network-object)# nat (inside,outside) dynamic MAPPED_IPS interface
```
Configuring Dynamic PAT (Hide)

This section describes how to configure network object NAT for dynamic PAT (hide). For more information, see the “Dynamic PAT” section on page 27-10.

Guidelines

For a PAT pool:

- If available, the real source port number is used for the mapped port. However, if the real port is not available, by default the mapped ports are chosen from the same range of ports as the real port number: 0 to 511, 512 to 1023, and 1024 to 65535.

For round robin for a PAT pool:

- If a host has an existing connection, then subsequent connections from that host will likely use different PAT addresses for each connection because of the round robin allocation. In this case, you may have problems when accessing two websites that exchange information about the host, for example an e-commerce site and a payment site. When these sites see two different IP addresses for what is supposed to be a single host, the transaction may fail.
## Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1**  
(Optional)  
Network object:  
`object network obj_name`  
(host ip_address | range ip_address_1 ip_address_2)  
Network object group:  
`object-group network grp_name`  
(network-object (object net_obj_name | host ip_address) |  
group-object grp_obj_name)  
Example:  
hostname(config)# object network PAT_POOL1  
hostname(config-network-object)# range 10.5.1.80 10.7.1.80  
hostname(config)# object network PAT_POOL2  
hostname(config-network-object)# range 10.9.1.1 10.10.1.1  
hostname(config)# object network PAT_IP  
hostname(config-network-object)# host 10.5.1.79  
hostname(config-network-object)# object-group network PAT_POOLS  
hostname(config-network-object)# network-object object PAT_POOL1  
hostname(config-network-object)# network-object object PAT_POOL2  
hostname(config-network-object)# network-object object PAT_IP  | Specify the **mapped** address(es) (that you want to translate to). You can configure a single address or, for a PAT pool, multiple addresses. Configure a network object or network object group. A network object group can contain objects and/or inline addresses. Alternatively, you can skip this step if you want to enter a single IP address as an inline value for the `nat` command or if you want to use the interface address by specifying the `interface` keyword. For mapped addresses used as a PAT pool, all addresses in the object or group, including ranges, are used as PAT addresses. **Note** The object or group cannot contain a subnet. See the “Guidelines and Limitations” section on page 28-2 for information about disallowed mapped IP addresses. For more information about configuring a network object or group, see the “Configuring Objects” section on page 12-3. |
| **Step 2**  
`object network obj_name`  
Example:  
hostname(config)# object network my-host-obj1  | Configures a network object for which you want to configure NAT, or enters object network configuration mode for an existing network object. |
| **Step 3**  
(host ip_address | subnet subnet_address netmask | range ip_address_1 ip_address_2)  
Example:  
hostname(config-network-object)# range 10.1.1.1 10.1.1.90  | If you are creating a new network object, defines the **real** IP address(es) that you want to translate. |
Configuring Network Object NAT

Step 4

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>nat</strong> [(real_ifc,mapped_ifc)] dynamic (mapped_inline_host_ip</td>
<td>mapped_obj</td>
</tr>
</tbody>
</table>

- **Interfaces**—(Required for transparent mode) Specify the real and mapped interfaces. Be sure to include the parentheses in your command. In routed mode, if you do not specify the real and mapped interfaces, all interfaces are used; you can also specify the keyword **any** for one or both of the interfaces.

  - **Mapped IP address**—You can specify the mapped IP address as:
    - An inline host address.
    - An existing network object that is defined as a host address (see **Step 1**).
    - **pat-pool**—An existing network object or group that contains multiple addresses.
    - **interface**—(Routed mode only) The IP address of the mapped interface is used as the mapped address. For this option, you must configure a specific interface for the **mapped_ifc**. You must use this keyword when you want to use the interface IP address; you cannot enter it inline or as an object.

  - For a PAT pool, you can specify one or more of the following options:
    - **Round robin**—The **round-robin** keyword enables round-robin address allocation for a PAT pool. Without round robin, by default all ports for a PAT address will be allocated before the next PAT address is used. The round-robin method assigns an address/port from each PAT address in the pool before returning to use the first address again, and then the second address, and so on.

    (continued)

  - **Interface PAT fallback**—(Optional) The **interface** keyword enables interface PAT fallback when entered after a primary PAT address. After the primary PAT address(es) are used up, then the IP address of the mapped interface is used. For this option, you must configure a specific interface for the **mapped_ifc**. (You cannot specify **interface** in transparent mode).

  - **DNS**—(Optional) The **dns** keyword translates DNS replies. Be sure DNS inspection is enabled (it is enabled by default). See the “DNS and NAT” section on page 27-24 for more information.

(continued)
Examples

The following example configures dynamic PAT that hides the 192.168.2.0 network behind address 10.2.2.2:

```
hostname(config)# object network my-inside-net
hostname(config-network-object)# subnet 192.168.2.0 255.255.255.0
hostname(config-network-object)# nat (inside,outside) dynamic 10.2.2.2
```

The following example configures dynamic PAT that hides the 192.168.2.0 network behind the outside interface address:

```
hostname(config)# object network my-inside-net
hostname(config-network-object)# subnet 192.168.2.0 255.255.255.0
hostname(config-network-object)# nat (inside,outside) dynamic interface
```

Configuring Static NAT or Static NAT-with-Port-Translation

This section describes how to configure a static NAT rule using network object NAT. For more information, see the “Static NAT” section on page 27-3.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>(Optional)</td>
</tr>
<tr>
<td>Network object:</td>
<td>To specify the mapped addresses (that you want to translate to), configure a network object or network object group. A network object group can contain objects and/or inline addresses. Alternatively, you can skip this step if you want to enter the IP addresses as an inline value for the nat command or if you want to use the interface address (for static NAT-with-port-translation) by specifying the interface keyword.</td>
</tr>
<tr>
<td><code>object network obj_name</code></td>
<td>Configures a network object for which you want to configure NAT, or enters object network configuration mode for an existing network object.</td>
</tr>
<tr>
<td><code>object network obj_name</code></td>
<td>For more information about configuring a network object or group, see the “Configuring Objects” section on page 12-3.</td>
</tr>
</tbody>
</table>

Example:

```
hostname(config)# object network MAPPED_IPS
hostname(config-network-object)# subnet 10.1.1.0 255.255.255.0
```

<table>
<thead>
<tr>
<th>Step 2</th>
<th>object network obj_name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configures a network object for which you want to configure NAT, or enters object network configuration mode for an existing network object.</td>
<td></td>
</tr>
</tbody>
</table>

Example:

```
hostname(config)# object network my-host-obj1
```
## Configuring Network Object NAT

**Step 3**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>(host ip_address</td>
<td>subnet subnet_address netmask</td>
</tr>
</tbody>
</table>

**Example:**

```
hostname(config-network-object)# subnet 10.2.1.0 255.255.255.0
```
Chapter 28 Configuring Network Object NAT

Step 4

**nat** \[(real_ifc,mapped_ifc)\] **static**
(mapped_inline_ip | mapped_obj | interface) [dns | service (tcp | udp)
real_port mapped_port] [no-proxy-arp]

**Command**

**Purpose**

Configures static NAT for the object IP addresses.

**Note** You can only define a single NAT rule for a given object. See the “Additional Guidelines” section on page 28-2.

See the following guidelines:

- **Interfaces**—(Required for transparent mode) Specify the real and mapped interfaces. Be sure to include the parentheses in your command. In routed mode, if you do not specify the real and mapped interfaces, all interfaces are used; you can also specify the keyword **any** for one or both of the interfaces.

- **Mapped IP Addresses**—You can specify the mapped IP address as:
  - An inline IP address. The netmask or range for the mapped network is the same as that of the real network. For example, if the real network is a host, then this address will be a host address. In the case of a range, then the mapped addresses include the same number of addresses as the real range. For example, if the real address is defined as a range from 10.1.1.1 through 10.1.1.6, and you specify 172.20.1.1 as the mapped address, then the mapped range will include 172.20.1.1 through 172.20.1.6.
  - An existing network object or group (see Step 1).
  - **interface**—(Static NAT-with-port-translation only; routed mode) For this option, you must configure a specific interface for the **mapped_ifc**. Be sure to also configure the **service** keyword.

Typically, you configure the same number of mapped addresses as real addresses for a one-to-one mapping. You can, however, have a mismatched number of addresses. For more information, see the “Static NAT” section on page 27-3.

- **DNS**—(Optional) The **dns** keyword translates DNS replies. Be sure DNS inspection is enabled (it is enabled by default). See the “DNS and NAT” section on page 27-24 for more information. This option is not available if you specify the **service** keyword.

- **Port translation**—(Static NAT-with-port-translation only) Specify **tcp** or **udp** and the real and mapped ports. You can enter either a port number or a well-known port name (such as **ftp**).

- **No Proxy ARP**—(Optional) Specify **no-proxy-arp** to disable proxy ARP for incoming packets to the mapped IP addresses. See the “Mapped Addresses and Routing” section on page 27-22 for more information.

Example:

```
hostname(config-network-object)# nat (inside,outside) static MAPPED_IPS service tcp 80 8080
```
Examples

The following example configures static NAT for the real host 10.1.1.1 on the inside to 10.2.2.2 on the outside with DNS rewrite enabled.

```
hostname(config)# object network my-host-obj1
hostname(config-network-object)# host 10.1.1.1
hostname(config-network-object)# nat (inside,outside) static 10.2.2.2 dns
```

The following example configures static NAT for the real host 10.1.1.1 on the inside to 2.2.2.2 on the outside using a mapped object.

```
hostname(config)# object network my-mapped-obj
hostname(config-network-object)# host 10.2.2.2
hostname(config-network-object)# object network my-host-obj1
hostname(config-network-object)# host 10.1.1.1
hostname(config-network-object)# nat (inside,outside) static my-mapped-obj
```

The following example configures static NAT-with-port-translation for 10.1.1.1 at TCP port 21 to the outside interface at port 2121.

```
hostname(config)# object network my-ftp-server
hostname(config-network-object)# host 10.1.1.1
hostname(config-network-object)# nat (inside,outside) static interface service tcp 21 2121
```

Configuring Identity NAT

This section describes how to configure an identity NAT rule using network object NAT. For more information, see the “Identity NAT” section on page 27-11.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>(Optional)</td>
</tr>
<tr>
<td>object network obj_name</td>
<td>For the mapped addresses (which will be the same as the real addresses), configure a network object. Alternatively, you can skip this step if you want to enter the IP addresses as an inline value for the nat command. For more information about configuring a network object, see the “Configuring Objects” section on page 12-3.</td>
</tr>
<tr>
<td>host ip_address</td>
<td></td>
</tr>
<tr>
<td>subnet subnet_address netmask</td>
<td></td>
</tr>
<tr>
<td>range ip_address_1 ip_address_2</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# object network MAPPED_IPS</td>
<td></td>
</tr>
<tr>
<td>hostname(config-network-object)# subnet 10.1.1.0 255.255.255.0</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td>object network obj_name</td>
<td>Configures a network object for which you want to perform identity NAT, or enters object network configuration mode for an existing network object.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# object network my-host-obj1</td>
<td></td>
</tr>
</tbody>
</table>
Chapter 28      Configuring Network Object NAT

Example
The following example maps a host address to itself using an inline mapped address:

```
hostname(config)# object network my-host-obj1
hostname(config-network-object)# host 10.1.1.1
hostname(config-network-object)# nat (inside,outside) static MAPPED_IPS
```

Note
You can only define a single NAT rule for a given object. See the “Additional Guidelines” section on page 28-2.
The following example maps a host address to itself using a network object:

```
hostname(config)# object network my-host-obj1-identity
hostname(config-network-object)# host 10.1.1.1
hostname(config-network-object)# object network my-host-obj1
hostname(config-network-object)# host 10.1.1.1
hostname(config-network-object)# nat (inside,outside) static my-host-obj1-identity
```

### Monitoring Network Object NAT

To monitor object NAT, enter one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show nat</td>
<td>Shows NAT statistics, including hits for each NAT rule.</td>
</tr>
<tr>
<td>show nat pool</td>
<td>Shows NAT pool statistics, including the addresses and ports allocated, and how many times they were allocated.</td>
</tr>
<tr>
<td>show running-config nat</td>
<td>Shows the NAT configuration.</td>
</tr>
</tbody>
</table>

**Note**: You cannot view the NAT configuration using the `show running-config object` command. You cannot reference objects or object groups that have not yet been created in `nat` commands. To avoid forward or circular references in `show` command output, the `show running-config` command shows the `object` command two times: first, where the IP address(es) are defined; and later, where the `nat` command is defined. This command output guarantees that objects are defined first, then object groups, and finally NAT. For example:

```
hostname# show running-config
...
object network obj1
    range 192.168.49.1 192.150.49.100
object network obj2
    object 192.168.49.100
object network network-1
    subnet <network-1>
object network network-2
    subnet <network-2>
object-group network pool
    network-object object obj1
    network-object object obj2
...
object network network-1
    nat (inside,outside) dynamic pool
object network network-2
    nat (inside,outside) dynamic pool
```

show xlate | Shows current NAT session information. |
Configuration Examples for Network Object NAT

This section includes the following configuration examples:

- Providing Access to an Inside Web Server (Static NAT), page 28-15
- NAT for Inside Hosts (Dynamic NAT) and NAT for an Outside Web Server (Static NAT), page 28-16
- Inside Load Balancer with Multiple Mapped Addresses (Static NAT, One-to-Many), page 28-17
- Single Address for FTP, HTTP, and SMTP (Static NAT-with-Port-Translation), page 28-18
- DNS Server on Mapped Interface, Web Server on Real Interface (Static NAT with DNS Modification), page 28-19
- DNS Server and Web Server on Mapped Interface, Web Server is Translated (Static NAT with DNS Modification), page 28-21

Providing Access to an Inside Web Server (Static NAT)

The following example performs static NAT for an inside web server. The real address is on a private network, so a public address is required. Static NAT is necessary so hosts can initiate traffic to the web server at a fixed address. (See Figure 28-1).

Figure 28-1 Static NAT for an Inside Web Server

![Diagram of static NAT for an inside web server]

Step 1 Create a network object for the internal web server:
```
hostname(config)# object network myWebServ
```

Step 2 Define the web server address:
```
hostname(config-network-object)# host 10.1.2.27
```
Step 3  Configure static NAT for the object:

```
hostname(config-network-object)# nat (inside,outside) static 209.165.201.10
```

**NAT for Inside Hosts (Dynamic NAT) and NAT for an Outside Web Server (Static NAT)**

The following example configures dynamic NAT for inside users on a private network when they access the outside. Also, when inside users connect to an outside web server, that web server address is translated to an address that appears to be on the inside network. (See Figure 28-2).

**Figure 28-2  Dynamic NAT for Inside, Static NAT for Outside Web Server**

---

**Step 1**  Create a network object for the dynamic NAT pool to which you want to translate the inside addresses:

```
hostname(config)# object network myNatPool
hostname(config-network-object)# range 209.165.201.20 209.165.201.30
```

**Step 2**  Create a network object for the inside network:

```
hostname(config)# object network myInsNet
hostname(config-network-object)# subnet 10.1.2.0 255.255.255.0
```

**Step 3**  Enable dynamic NAT for the inside network:

```
hostname(config-network-object)# nat (inside,outside) dynamic myNatPool
```
Step 4  Create a network object for the outside web server:
hostname(config)# object network myWebServ

Step 5  Define the web server address:
hostname(config-network-object)# host 209.165.201.12

Step 6  Configure static NAT for the web server:
hostname(config-network-object)# nat (outside,inside) static 10.1.2.20

Inside Load Balancer with Multiple Mapped Addresses (Static NAT, One-to-Many)

The following example shows an inside load balancer that is translated to multiple IP addresses. When an outside host accesses one of the mapped IP addresses, it is untranslated to the single load balancer address. Depending on the URL requested, it redirects traffic to the correct web server. (See Figure 28-3).

Figure 28-3  Static NAT with One-to-Many for an Inside Load Balancer

Step 1  Create a network object for the addresses to which you want to map the load balancer:
Step 2  Create a network object for the load balancer:
hostname(config)# object network myLBHost

Step 3  Define the load balancer address:
hostname(config-network-object)# host 10.1.2.27

Step 4  Configure static NAT for the load balancer:
hostname(config-network-object)# nat (inside,outside) static myPublicIPs

---

**Single Address for FTP, HTTP, and SMTP (Static NAT-with-Port-Translation)**

The following static NAT-with-port-translation example provides a single address for remote users to access FTP, HTTP, and SMTP. These servers are actually different devices on the real network, but for each server, you can specify static NAT-with-port-translation rules that use the same mapped IP address, but different ports. (See Figure 28-4.)

**Figure 28-4   Static NAT-with-Port-Translation**

Step 1  Create a network object for the FTP server address:
hostname(config)# object network FTP_SERVER

---

**Table**: Host and Outside Addresses and Port Translations

<table>
<thead>
<tr>
<th>Host Address</th>
<th>Outside Address</th>
<th>Port Translation</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.1.2.27</td>
<td>209.165.201.3:21</td>
<td>10.1.2.27</td>
</tr>
<tr>
<td>10.1.2.28</td>
<td>209.165.201.3:80</td>
<td>10.1.2.28</td>
</tr>
<tr>
<td>10.1.2.29</td>
<td>209.165.201.3:25</td>
<td>10.1.2.29</td>
</tr>
</tbody>
</table>

**Server Addresses**:
- FTP server: 10.1.2.27
- HTTP server: 10.1.2.28
- SMTP server: 10.1.2.29
Chapter 28  Configuring Network Object NAT

Configuration Examples for Network Object NAT

Step 2  Define the FTP server address, and configure static NAT with identity port translation for the FTP server:

```
hostname(config-network-object)# host 10.1.2.27
hostname(config-network-object)# nat (inside, outside) static 209.165.201.3 service tcp ftp ftp
```

Step 3  Create a network object for the HTTP server address:

```
hostname(config)# object network HTTP_SERVER
```

Step 4  Define the HTTP server address, and configure static NAT with identity port translation for the HTTP server:

```
hostname(config-network-object)# host 10.1.2.28
hostname(config-network-object)# nat (inside, outside) static 209.165.201.3 service tcp http http
```

Step 5  Create a network object for the SMTP server address:

```
hostname(config)# object network SMTP_SERVER
```

Step 6  Define the SMTP server address, and configure static NAT with identity port translation for the SMTP server:

```
hostname(config-network-object)# host 10.1.2.29
hostname(config-network-object)# nat (inside, outside) static 209.165.201.3 service tcp smtp smtp
```

DNS Server on Mapped Interface, Web Server on Real Interface (Static NAT with DNS Modification)

For example, a DNS server is accessible from the outside interface. A server, ftp.cisco.com, is on the inside interface. You configure the ASASM to statically translate the ftp.cisco.com real address (10.1.3.14) to a mapped address (209.165.201.10) that is visible on the outside network. (See Figure 28-5.) In this case, you want to enable DNS reply modification on this static rule so that inside users who have access to ftp.cisco.com using the real address receive the real address from the DNS server, and not the mapped address.
When an inside host sends a DNS request for the address of ftp.cisco.com, the DNS server replies with the mapped address (209.165.201.10). The ASASM refers to the static rule for the inside server and translates the address inside the DNS reply to 10.1.3.14. If you do not enable DNS reply modification, then the inside host attempts to send traffic to 209.165.201.10 instead of accessing ftp.cisco.com directly.

*Figure 28-5  DNS Reply Modification*

```
Step 1 Create a network object for the FTP server address:
hostname(config)# object network FTP_SERVER

Step 2 Define the FTP server address, and configure static NAT with DNS modification:
hostname(config-network-object)# host 10.1.3.14
hostname(config-network-object)# nat (inside,outside) static 209.165.201.10 dns
```
**DNS Server and Web Server on Mapped Interface, Web Server is Translated (Static NAT with DNS Modification)**

Figure 28-6 shows a web server and DNS server on the outside. The ASASM has a static translation for the outside server. In this case, when an inside user requests the address for ftp.cisco.com from the DNS server, the DNS server responds with the real address, 209.165.20.10. Because you want inside users to use the mapped address for ftp.cisco.com (10.1.2.56) you need to configure DNS reply modification for the static translation.

**Figure 28-6  DNS Reply Modification Using Outside NAT**

![Diagram showing DNS reply modification](image)

---

**Step 1**  Create a network object for the FTP server address:

```
hostname(config)# object network FTP_SERVER
```

**Step 2**  Define the FTP server address, and configure static NAT with DNS modification:

```
hostname(config-network-object)# host 209.165.201.10
hostname(config-network-object)# nat (outside,inside) static 10.1.2.56 dns
```
# Feature History for Network Object NAT

Table 28-1 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network Object NAT</td>
<td>8.3(1)</td>
<td>Configures NAT for a network object IP address(es). We introduced or modified the following commands: nat (object network configuration mode), show nat, show xlate, show nat pool.</td>
</tr>
<tr>
<td>Identity NAT configurable proxy ARP and route lookup</td>
<td>8.4(2)/8.5(1)</td>
<td>In earlier releases for identity NAT, proxy ARP was disabled, and a route lookup was always used to determine the egress interface. You could not configure these settings. In 8.4(2) and later, the default behavior for identity NAT was changed to match the behavior of other static NAT configurations: proxy ARP is enabled, and the NAT configuration determines the egress interface (if specified) by default. You can leave these settings as is, or you can enable or disable them discretely. Note that you can now also disable proxy ARP for regular static NAT. When upgrading to 8.4(2) from 8.3(1), 8.3(2), and 8.4(1), all identity NAT configurations will now include the no-proxy-arp and route-lookup keywords, to maintain existing functionality. We modified the following commands: nat static [no-proxy-arp] [route-lookup].</td>
</tr>
<tr>
<td>PAT pool and round robin address assignment</td>
<td>8.4(2)/8.5(1)</td>
<td>You can now specify a pool of PAT addresses instead of a single address. You can also optionally enable round-robin assignment of PAT addresses instead of first using all ports on a PAT address before using the next address in the pool. These features help prevent a large number of connections from a single PAT address from appearing to be part of a DoS attack and makes configuration of large numbers of PAT addresses easy. We modified the following commands: nat dynamic [pat-pool mapped_object [round-robin]].</td>
</tr>
</tbody>
</table>
Configuring Twice NAT

Twice NAT lets you identify both the source and destination address in a single rule. This chapter shows you how to configure twice NAT and includes the following sections:

- Information About Twice NAT, page 29-1
- Licensing Requirements for Twice NAT, page 29-2
- Prerequisites for Twice NAT, page 29-2
- Guidelines and Limitations, page 29-2
- Default Settings, page 29-3
- Configuring Twice NAT, page 29-3
- Monitoring Twice NAT, page 29-22
- Configuration Examples for Twice NAT, page 29-22
- Feature History for Twice NAT, page 29-26

**Note**
For detailed information about how NAT works, see Chapter 27, “Information About NAT.”

### Information About Twice NAT

Twice NAT lets you identify both the source and destination address in a single rule. Specifying both the source and destination addresses lets you specify that a source address should be translated to A when going to destination X, but be translated to B when going to destination Y, for example.

**Note**
For static NAT, the rule is bidirectional, so be aware that “source” and “destination” are used in commands and descriptions throughout this guide even though a given connection might originate at the “destination” address. For example, if you configure static NAT with port address translation, and specify the source address as a Telnet server, and you want all traffic going to that Telnet server to have the port translated from 2323 to 23, then in the command, you must specify the source ports to be translated (real: 23, mapped: 2323). You specify the source ports because you specified the Telnet server address as the source address.

The destination address is optional. If you specify the destination address, you can either map it to itself (identity NAT), or you can map it to a different address. The destination mapping is always a static mapping.
Twice NAT also lets you use service objects for static NAT-with-port-translation; network object NAT only accepts inline definition.

For detailed information about the differences between twice NAT and network object NAT, see the “How NAT is Implemented” section on page 27-16.

Twice NAT rules are added to section 1 of the NAT rules table, or if specified, section 3. For more information about NAT ordering, see the “NAT Rule Order” section on page 27-20.

### Licensing Requirements for Twice NAT

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

### Prerequisites for Twice NAT

- For both the real and mapped addresses, configure network objects or network object groups (the `object network` or `object-group network` command). Network object groups are particularly useful for creating a mapped address pool with discontinuous IP address ranges or multiple hosts or subnets. To create a network object or group, see the “Configuring Objects and Groups” section on page 12-1.

- For static NAT-with-port-translation, configure TCP or UDP service objects (the `object service` command). To create a service object, see the “Configuring a Service Object” section on page 12-4.

For specific guidelines for objects and groups, see the configuration section for the NAT type you want to configure. See also the “Guidelines and Limitations” section.

### Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

#### Context Mode Guidelines

Supported in single and multiple context mode.

#### Firewall Mode Guidelines

- Supported in routed and transparent firewall mode.
- In transparent mode, you must specify the real and mapped interfaces; you cannot use **any**.
- In transparent mode, you cannot configure interface PAT, because the transparent mode interfaces do not have IP addresses. You also cannot use the management IP address as a mapped address.

#### IPv6 Guidelines

Does not support IPv6.
Additional Guidelines

- If you change the NAT configuration, and you do not want to wait for existing translations to time out before the new NAT information is used, you can clear the translation table using the `clear xlate` command. However, clearing the translation table disconnects all current connections that use translations.

  **Note**  
  If you remove a dynamic NAT or PAT rule, and then add a new rule with mapped addresses that overlap the addresses in the removed rule, then the new rule will not be used until all connections associated with the removed rule time out or are cleared using the `clear xlate` command. This safeguard ensures that the same address is not assigned to multiple hosts.

- Objects and object groups used in NAT cannot be undefined; they must include IP addresses.
- You can use the same objects in multiple rules.
- The mapped IP address pool cannot include:
  - The mapped interface IP address. If you specify any interface for the rule, then all interface IP addresses are disallowed. For interface PAT (routed mode only), use the `interface` keyword instead of the IP address.
  - (Transparent mode) The management IP address.
  - (Dynamic NAT) The standby interface IP address when VPN is enabled.
  - Existing VPN pool addresses.

Default Settings

- By default, the rule is added to the end of section 1 of the NAT table.
- (Routed mode) The default real and mapped interface is Any, which applies the rule to all interfaces.
- (The default behavior for identity NAT has proxy ARP enabled, matching other static NAT rules. You can disable proxy ARP if desired.
- If you specify an optional interface, then the ASASM uses the NAT configuration to determine the egress interface. For identity NAT, the default behavior is to use the NAT configuration, but you have the option to always use a route lookup instead.

Configuring Twice NAT

This section describes how to configure twice NAT. This section includes the following topics:

- Configuring Dynamic NAT, page 29-4
- Configuring Dynamic PAT (Hide), page 29-8
- Configuring Static NAT or Static NAT-with-Port-Translation, page 29-13
- Configuring Identity NAT, page 29-18
## Configuring Dynamic NAT

This section describes how to configure twice NAT for dynamic NAT. For more information, see the “Dynamic NAT” section on page 27-8.

### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td>Network object:</td>
<td>Configure the <strong>real source</strong> addresses.</td>
</tr>
<tr>
<td><em>object network obj_name</em>&lt;br&gt; {host ip_address</td>
<td>subnet subnet_address netmask</td>
</tr>
<tr>
<td>Network object group:</td>
<td>If you want to translate all traffic, you can skip this step and specify the <em>any</em> keyword instead of creating an object or group.</td>
</tr>
<tr>
<td><em>object-group network grp_name</em>&lt;br&gt; {network-object {object net_obj_name</td>
<td>subnet_address netmask</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td>Network object:</td>
<td>Configure the <strong>mapped source</strong> addresses.</td>
</tr>
<tr>
<td><em>object network obj_name</em>&lt;br&gt; {range ip_address_1 ip_address_2}</td>
<td>You can configure either a network object or a network object group.</td>
</tr>
<tr>
<td>Network object group:</td>
<td>For dynamic NAT, you typically configure a larger group of addresses to be mapped to a smaller group. If a mapped network object contains both ranges and host IP addresses, then the ranges are used for dynamic NAT, and then the host IP addresses are used as a PAT fallback.</td>
</tr>
<tr>
<td><em>object-group network grp_name</em>&lt;br&gt; {network-object {object net_obj_name</td>
<td>host ip_address}</td>
</tr>
</tbody>
</table>

### Example:

**Step 1**

```bash
hostname(config)# object network MyInsNet
hostname(config-network-object)# subnet 10.1.1.0 255.255.255.0
```

**Step 2**

```bash
hostname(config)# object network NAT_POOL
hostname(config-network-object)# range 209.165.201.10 209.165.201.20
```
### Step 3 (Optional)

**Network object:**

```
object network obj_name
    (host ip_address | subnet subnet_address netmask | range ip_address_1 ip_address_2)
```

**Network object group:**

```
object-group network grp_name
    (network-object (object net_obj_name | subnet_address netmask | host ip_address) | group-object grp_obj_name)
```

**Example:**

```
hostname(config)# object network Server1
hostname(config-network-object)# host 209.165.201.8
```

### Step 4 (Optional)

**Network object:**

```
object network obj_name
    (host ip_address | subnet subnet_address netmask | range ip_address_1 ip_address_2)
```

**Network object group:**

```
object-group network grp_name
    (network-object (object net_obj_name | subnet_address netmask | host ip_address) | group-object grp_obj_name)
```

**Example:**

```
hostname(config)# object network Server1_mapped
hostname(config-network-object)# host 10.1.1.67
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 3</strong></td>
<td>Configure the <strong>real destination</strong> addresses. You can configure either a network object or a network object group. Although the main feature of twice NAT is the inclusion of the destination IP address, the destination address is optional. If you do specify the destination address, you can configure static translation for that address or just use identity NAT for it. You might want to configure twice NAT without a destination address to take advantage of some of the other qualities of twice NAT, including the use of network object groups for real addresses, or manually ordering of rules. For more information, see the “Main Differences Between Network Object NAT and Twice NAT” section on page 27-16.</td>
</tr>
<tr>
<td>Network object:</td>
<td></td>
</tr>
<tr>
<td><strong>object network obj_name</strong></td>
<td></td>
</tr>
<tr>
<td>(host ip_address</td>
<td>subnet subnet_address netmask</td>
</tr>
<tr>
<td><strong>Network object group:</strong></td>
<td></td>
</tr>
<tr>
<td><strong>object-group network grp_name</strong></td>
<td></td>
</tr>
<tr>
<td>(network-object (object net_obj_name</td>
<td>subnet_address netmask</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# object network Server1</td>
<td></td>
</tr>
<tr>
<td>hostname(config-network-object)# host 209.165.201.8</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>Configure the <strong>mapped destination</strong> addresses. The destination translation is always static. For identity NAT, you can skip this step and simply use the same object or group for both the real and mapped addresses. If you want to translate the destination address, you can configure either a network object or a network object group. The static mapping is typically one-to-one, so the real addresses have the same quantity as the mapped addresses. You can, however, have different quantities if desired. For more information, see the “Static NAT” section on page 27-3. For static interface NAT with port translation (routed mode only), you can skip this step and specify the <strong>interface</strong> keyword instead of a network object/group for the mapped address. For more information, see the “Static Interface NAT with Port Translation” section on page 27-5.</td>
</tr>
<tr>
<td>Network object:</td>
<td></td>
</tr>
<tr>
<td><strong>object network obj_name</strong></td>
<td></td>
</tr>
<tr>
<td>(host ip_address</td>
<td>subnet subnet_address netmask</td>
</tr>
<tr>
<td><strong>Network object group:</strong></td>
<td></td>
</tr>
<tr>
<td><strong>object-group network grp_name</strong></td>
<td></td>
</tr>
<tr>
<td>(network-object (object net_obj_name</td>
<td>subnet_address netmask</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# object network Server1_mapped</td>
<td></td>
</tr>
<tr>
<td>hostname(config-network-object)# host 10.1.1.67</td>
<td></td>
</tr>
</tbody>
</table>
Step 5

(Optional)

```
object service obj_name
    service {tcp | udp} destination
    operator port
```

**Example:**

```
hostname(config)# object service REAL_SVC
hostname(config-service-object)# service tcp destination eq 80
```

```
hostname(config)# object service MAPPED_SVC
hostname(config-service-object)# service tcp destination eq 8080
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Optional)</td>
<td>Configure service objects for:</td>
</tr>
<tr>
<td></td>
<td>• Destination real port</td>
</tr>
<tr>
<td></td>
<td>• Destination mapped port</td>
</tr>
</tbody>
</table>

Dynamic NAT does not support port translation. However, because the destination translation is always static, you can perform port translation for the destination port. A service object can contain both a source and destination port, but only the destination port is used in this case. If you specify the source port, it will be ignored. NAT only supports TCP or UDP. When translating a port, be sure the protocols in the real and mapped service objects are identical (both TCP or both UDP). For identity NAT, you can use the same service object for both the real and mapped ports. The “not equal” (neq) operator is not supported.
### Command

<table>
<thead>
<tr>
<th>Step 6</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **nat** 
[ (real_ifc,mapped_ifc)] 
[ line | (after-auto [line])] 
source dynamic 
{real_obj | any} 
{mapped_obj [interface]} 
[ destination static 
{mapped_obj | interface} real_obj] 
[ service mapped_svc_obj 
real_dest_svc_obj] [dns] [inactive] 
[ description desc] | Configure **dynamic NAT**. See the following guidelines: |

- **Interfaces**—(Required for transparent mode) Specify the real and mapped interfaces. Be sure to include the parentheses in your command. In routed mode, if you do not specify the real and mapped interfaces, all interfaces are used; you can also specify the keyword **any** for one or both of the interfaces.

- **Section and Line**—(Optional) By default, the NAT rule is added to the end of section 1 of the NAT table (see the “NAT Rule Order” section on page 27-20). If you want to add the rule into section 3 instead (after the network object NAT rules), then use the **after-auto** keyword. You can insert a rule anywhere in the applicable section using the **line** argument.

- **Source addresses:**
  - **Real**—Specify a network object, group, or the **any** keyword (see **Step 1**). Use the **any** keyword if you want to translate all traffic from the real interface to the mapped interface.
  - **Mapped**—Specify a different network object or group (see **Step 2**). You can optionally configure the following fallback method:

    Interface PAT fallback—(Routed mode only) The **interface** keyword enables interface PAT fallback. After the mapped IP addresses are used up, then the IP address of the mapped interface is used. For this option, you must configure a specific interface for the **mapped_ifc**.

---

Example:

```
hostname(config)# nat (inside,outside) 
source dynamic MyInsNet NAT_POOL 
destination static Server1_mapped Server1 
service MAPPED_SVC REAL_SVC
```
Configuring Dynamic PAT (Hide)

This section describes how to configure twice NAT for dynamic PAT (hide). For more information, see the “Dynamic PAT” section on page 27-10.

Guidelines

For a PAT pool:

- If available, the real source port number is used for the mapped port. However, if the real port is not available, by default the mapped ports are chosen from the same range of ports as the real port number: 0 to 511, 512 to 1023, and 1024 to 65535.

For round robin for a PAT pool:

- If a host has an existing connection, then subsequent connections from that host will likely use different PAT addresses for each connection because of the round robin allocation. In this case, you may have problems when accessing two websites that exchange information about the host, for example an e-commerce site and a payment site. When these sites see two different IP addresses for what is supposed to be a single host, the transaction may fail.
### Chapter 29  Configuring Twice NAT

**Detailed Steps**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Network object:</strong></td>
<td></td>
</tr>
<tr>
<td><code>object network obj_name</code></td>
<td></td>
</tr>
<tr>
<td>(`host ip_address</td>
<td>subnet subnet_address netmask</td>
</tr>
<tr>
<td><strong>Network object group:</strong></td>
<td></td>
</tr>
<tr>
<td><code>object-group network grp_name</code></td>
<td></td>
</tr>
<tr>
<td>(`network-object (object net_obj_name</td>
<td>subnet_address netmask</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# object network MyInsNet</td>
<td></td>
</tr>
<tr>
<td>hostname(config-network-object)# subnet 10.1.1.0 255.255.255.0</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Network object:</strong></td>
<td></td>
</tr>
<tr>
<td><code>object network obj_name</code></td>
<td></td>
</tr>
<tr>
<td>(`host ip_address</td>
<td>range ip_address_1 ip_address_2`)</td>
</tr>
<tr>
<td><strong>Network object group:</strong></td>
<td></td>
</tr>
<tr>
<td><code>object-group network grp_name</code></td>
<td></td>
</tr>
<tr>
<td>(`network-object (object net_obj_name</td>
<td>host ip_address)</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# object network PAT_POOL1</td>
<td></td>
</tr>
<tr>
<td>hostname(config-network-object)# range 10.5.1.80 10.7.1.80</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# object network PAT_POOL2</td>
<td></td>
</tr>
<tr>
<td>hostname(config-network-object)# range 10.9.1.1 10.10.1.1</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# object network PAT.IP</td>
<td></td>
</tr>
<tr>
<td>hostname(config-network-object)# host 10.5.1.79</td>
<td></td>
</tr>
<tr>
<td>hostname(config-network-object)# object-group network PAT.POOLS</td>
<td></td>
</tr>
<tr>
<td>object-PAT_POOL1</td>
<td>network-object object PAT_POOL1</td>
</tr>
</tbody>
</table>

**Command Purpose**

**Step 1**

Configure the **real source** addresses.

You can configure either a network object or a network object group. For more information, see the “Configuring Objects” section on page 12-3.

If you want to translate all traffic, you can skip this step and specify the **any** keyword instead of creating an object or group.

**Step 2**

Specify the **mapped** address(es) (that you want to translate to).

You can configure a single address or, for a PAT pool, multiple addresses. Configure a network object or network object group. A network object group can contain objects and/or inline addresses. Alternatively, you can skip this step if you want to enter a single IP address as an inline value for the **nat** command or if you want to use the interface address by specifying the **interface** keyword.

For mapped addresses used as a PAT pool, all addresses in the object or group, including ranges, are used as PAT addresses.

**Note**  The object or group cannot contain a subnet.

See the “Guidelines and Limitations” section on page 29-2 for information about disallowed mapped IP addresses.

For more information about configuring a network object or group, see the “Configuring Objects” section on page 12-3.
Chapter 29      Configuring Twice NAT

Configuring Twice NAT

Step 3
(Optional)

Network object:

```
object network obj_name
{host ip_address | subnet
 subnet_address netmask | range
 ip_address_1 ip_address_2)
```

Network object group:

```
object-group network grp_name
{network-object {object net_obj_name |
 subnet_address netmask |
 host ip_address) | |
 group-object grp_obj_name)
```

Example:

```bash
hostname(config)# object network Server1
hostname(config-network-object)# host 209.165.201.8
```

Step 4
(Optional)

Network object:

```
object network obj_name
{host ip_address | subnet
 subnet_address netmask | range
 ip_address_1 ip_address_2)
```

Network object group:

```
object-group network grp_name
{network-object {object net_obj_name |
 subnet_address netmask |
 host ip_address) | |
 group-object grp_obj_name)
```

Example:

```bash
hostname(config)# object network Server1_mapped
hostname(config-network-object)# host 10.1.1.67
```

Command Purpose

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 3**
(Optional)

Network object:

- Configure the **real destination** addresses.

You can configure either a network object or a network object group.

Although the main feature of twice NAT is the inclusion of the destination IP address, the destination address is optional. If you do specify the destination address, you can configure static translation for that address or just use identity NAT for it. You might want to configure twice NAT without a destination address to take advantage of some of the other qualities of twice NAT, including the use of network object groups for real addresses, or manually ordering of rules. For more information, see the “Main Differences Between Network Object NAT and Twice NAT” section on page 27-16.

Network object group:

- Configure the **mapped destination** addresses.

The destination translation is always static. For identity NAT, you can skip this step and simply use the same object or group for both the real and mapped addresses.

If you want to translate the destination address, you can configure either a network object or a network object group. The static mapping is typically one-to-one, so the real addresses have the same quantity as the mapped addresses. You can, however, have different quantities if desired. For more information, see the “Static NAT” section on page 27-3.

For static interface NAT with port translation (routed mode only), you can skip this step and specify the **interface** keyword instead of a network object/group for the mapped address. For more information, see the “Static Interface NAT with Port Translation” section on page 27-5.
Step 5
(Optional)

```
object service obj_name
  service {tcp | udp} destination operator port
```

Example:
```
hostname(config)# object service REAL_SVC
hostname(config-service-object)# service tcp destination eq 80
```

```
hostname(config)# object service MAPPED_SVC
hostname(config-service-object)# service tcp destination eq 8080
```

Step 6

```
nat [\{real_ifc,mapped_ifc\}]
  [\{line | \{after-auto [\{line\}\}\}\}
  source dynamic \{real-obj | any\}
  \{mapped_obj \{interface\} | \{pat-pool mapped_obj \{round-robin\} \{interface\} | \{interface\}\}
  \{destination static \{mapped_obj \{interface\} | interface\}
  \{service mapped_dest_svc_obj \{interface\} | interface\}
  real_obj]
  mapped_obj \{service mapped_dest_svc_obj \{interface\} | interface\}
  [dns] [inactive]
  [description desc]
```

Example:
```
hostname(config)# nat \{inside, outside\}
source dynamic MyInsNet interface destination static Server1 Server1
description Interface PAT for inside addresses when going to server 1
```

Configure service objects for:

- **Destination real port**
- **Destination mapped port**

Dynamic PAT does not support additional port translation. However, because the destination translation is always static, you can perform port translation for the destination port. A service object can contain both a source and destination port, but only the destination port is used in this case. If you specify the source port, it will be ignored. NAT only supports TCP or UDP. When translating a port, be sure the protocols in the real and mapped service objects are identical (both TCP or both UDP). For identity NAT, you can use the same service object for both the real and mapped ports. The “not equal” (neq) operator is not supported.

Configures dynamic PAT (hide). See the following guidelines:

- **Interfaces**—(Required for transparent mode) Specify the real and mapped interfaces. Be sure to include the parentheses in your command. In routed mode, if you do not specify the real and mapped interfaces, all interfaces are used; you can also specify the keyword any for one or both of the interfaces.

- **Section and Line**—(Optional) By default, the NAT rule is added to the end of section 1 of the NAT table (see the “NAT Rule Order” section on page 27-20). If you want to add the rule into section 3 instead (after the network object NAT rules), then use the after-auto keyword. You can insert a rule anywhere in the applicable section using the line argument.

- **Source addresses**:
  - **Real**—Specify a network object, group, or the any keyword (see Step 1). Use the any keyword if you want to translate all traffic from the real interface to the mapped interface.
  - **Mapped**—Configure one of the following:
    - Network object—Specify a network object that contains a host address (see Step 2).
    - pat-pool—Specify the pat-pool keyword and a network object or group that contains multiple addresses (see Step 2).
    - interface—(Routed mode only) Specify the interface keyword alone to only use interface PAT. When specified with a PAT pool or network object, the interface keyword enables interface PAT fallback. After the PAT IP addresses are used up, then the IP address of the mapped interface is used. For this option, you must configure a specific interface for the mapped_ifc.

(continued)
For a PAT pool, you can specify one or more of the following options:

-- Round robin—The **round-robin** keyword enables round-robin address allocation for a PAT pool. Without round robin, by default all ports for a PAT address will be allocated before the next PAT address is used. The round-robin method assigns an address/port from each PAT address in the pool before returning to use the first address again, and then the second address, and so on.

**Destination addresses (Optional):**

- Mapped—Specify a network object or group, or for static interface NAT with port translation only (routed mode), specify the **interface** keyword (see Step 4). If you specify **interface**, be sure to also configure the **service** keyword. For this option, you must configure a specific interface for the **real_ifc**. See the “Static Interface NAT with Port Translation” section on page 27-5 for more information.

- Real—Specify a network object or group (see Step 3). For identity NAT, simply use the same object or group for both the real and mapped addresses.

**Destination port—** (Optional) Specify the **service** keyword along with the real and mapped service objects (see Step 5). For identity port translation, simply use the same service object for both the real and mapped ports.

**DNS—** (Optional; for a source-only rule) The **dns** keyword translates DNS replies. Be sure DNS inspection is enabled (it is enabled by default). You cannot configure the **dns** keyword if you configure a **destination** address. See the “DNS and NAT” section on page 27-24 for more information.

**Inactive—** (Optional) To make this rule inactive without having to remove the command, use the **inactive** keyword. To reactivate it, reenter the whole command without the **inactive** keyword.

**Description—** (Optional) Provide a description up to 200 characters using the **description** keyword.
# Configuring Static NAT or Static NAT-with-Port-Translation

This section describes how to configure a static NAT rule using twice NAT. For more information about static NAT, see the “Static NAT” section on page 27-3.

## Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Configure <strong>the real source</strong> addresses. You can configure either a network object or a network object group. For more information, see the “Configuring Objects” section on page 12-3.</td>
</tr>
<tr>
<td><strong>Network object:</strong></td>
<td><strong>object network</strong> obj_name ({\text{host} \ ip_address</td>
</tr>
<tr>
<td><strong>Network object group:</strong></td>
<td><strong>object-group network</strong> grp_name ({\text{network-object} \ (\text{object} \ net_obj_name</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname(config)# object network MyInsNet hostname(config-network-object)# subnet 10.1.1.0 255.255.255.0</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Configure the <strong>mapped source</strong> addresses. You can configure either a network object or a network object group. For static NAT, the mapping is typically one-to-one, so the real addresses have the same quantity as the mapped addresses. You can, however, have different quantities if desired. For more information, see the “Static NAT” section on page 27-3.</td>
</tr>
<tr>
<td><strong>Network object:</strong></td>
<td><strong>object network</strong> obj_name ({\text{host} \ ip_address</td>
</tr>
<tr>
<td><strong>Network object group:</strong></td>
<td><strong>object-group network</strong> grp_name ({\text{network-object} \ (\text{object} \ net_obj_name</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname(config)# object network MyInsNet_mapped hostname(config-network-object)# subnet 192.168.1.0 255.255.255.0</td>
</tr>
</tbody>
</table>

For static interface NAT with port translation (routed mode only), you can skip this step and specify the **interface** keyword instead of a network object/group for the mapped address. For more information, see the “Static Interface NAT with Port Translation” section on page 27-5.

See the “Guidelines and Limitations” section on page 29-2 for information about disallowed mapped IP addresses.
**Chapter 29      Configuring Twice NAT**

**Step 3**  
(Optional)

**Network object:**

```
object network obj_name
   {host ip_address | subnet subnet_address netmask | range ip_address_1 ip_address_2}
```

**Network object group:**

```
object-group network grp_name
   {network-object {object net_obj_name | subnet_address netmask | host ip_address | group-object grp_obj_name}}
```

**Example:**

```
hostname(config)# object network Server1
hostname(config-network-object)# host 209.165.201.8
```

**Step 4**  
(Optional)

**Network object:**

```
object network obj_name
   {host ip_address | subnet subnet_address netmask | range ip_address_1 ip_address_2}
```

**Network object group:**

```
object-group network grp_name
   {network-object {object net_obj_name | subnet_address netmask | host ip_address | group-object grp_obj_name}}
```

**Example:**

```
hostname(config)# object network Server1_mapped
hostname(config-network-object)# host 10.1.1.67
```

**Command**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuring the <strong>real destination</strong> addresses. You can configure either a network object or a network object group. Although the main feature of twice NAT is the inclusion of the destination IP address, the destination address is optional. If you do specify the destination address, you can configure static translation for that address or just use identity NAT for it. You might want to configure twice NAT without a destination address to take advantage of some of the other qualities of twice NAT, including the use of network object groups for real addresses, or manually ordering of rules. For more information, see the “Main Differences Between Network Object NAT and Twice NAT” section on page 27-16.</td>
<td></td>
</tr>
</tbody>
</table>

| Configure the **mapped destination** addresses. The destination translation is always static. For identity NAT, you can skip this step and simply use the same object or group for both the real and mapped addresses. If you want to translate the destination address, you can configure either a network object or a network object group. The static mapping is typically one-to-one, so the real addresses have the same quantity as the mapped addresses. You can, however, have different quantities if desired. For more information, see the “Static NAT” section on page 27-3. For static interface NAT with port translation (routed mode only), you can skip this step and specify the interface keyword instead of a network object/group for the mapped address. For more information, see the “Static Interface NAT with Port Translation” section on page 27-5. |

---

*Cisco ASA Services Module CLI Configuration Guide*
Step 5  
(Optional)

```
object service obj_name
  service {tcp | udp} [source operator port] [destination operator port]
```

Example:
```
hostname(config)# object service REAL_SRC_SVC
hostname(config-service-object)# service tcp source eq 80

hostname(config)# object service MAPPED_SRC_SVC
hostname(config-service-object)# service tcp source eq 8080
```

Configure service objects for:

- **Source or destination real port**
- **Source or destination mapped port**

A service object can contain both a source and destination port; however, you should specify *either* the source *or* the destination port for both service objects. You should only specify *both* the source and destination ports if your application uses a fixed source port (such as some DNS servers); but fixed source ports are rare. NAT only supports TCP or UDP. When translating a port, be sure the protocols in the real and mapped service objects are identical (both TCP or both UDP). For identity NAT, you can use the same service object for both the real and mapped ports. The “not equal” (*neq*) operator is not supported.

For example, if you want to translate the port for the source host, then configure the source service.
### Step 6

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>nat [(real_ifc,mapped_ifc)]</td>
<td>Configures static NAT. See the following guidelines:</td>
</tr>
<tr>
<td>[line</td>
<td>(after-object [line])]</td>
</tr>
<tr>
<td>source static real_ob</td>
<td>• Section and Line—(Optional) By default, the NAT rule is added to the end of section 1 of the NAT table. See the “NAT Rule Order” section on page 27-20 for more information about sections. If you want to add the rule into section 3 instead (after the network object NAT rules), then use the after-auto keyword. You can insert a rule anywhere in the applicable section using the line argument.</td>
</tr>
<tr>
<td>[mapped_obj</td>
<td>interface]</td>
</tr>
<tr>
<td>destination static (mapped_obj</td>
<td>interface) real_obj</td>
</tr>
<tr>
<td>[service real_src_mapped_dest_svc_obj mapped_src_real_dest_svc_obj]</td>
<td></td>
</tr>
<tr>
<td>[no-proxy-arp</td>
<td>inactive]</td>
</tr>
<tr>
<td>[description desc]</td>
<td>– Mapped—Specify a network object or group, or for static interface NAT with port translation only, specify the interface keyword (see Step 4). If you specify interface, be sure to also configure the service keyword (in this case, the service objects should include only the destination port). For this option, you must configure a specific interface for the real_ifc.</td>
</tr>
<tr>
<td></td>
<td>– Real—Specify a network object or group (see Step 3). For identity NAT, simply use the same object or group for both the real and mapped addresses.</td>
</tr>
</tbody>
</table>

Example:
```
hostname(config)# nat (inside,dmz) source static MyInsNet MyInsNet_mapped
destination static Server1 Server1 service REAL_SRC_SVC MAPPED_SRC_SVC
```
Chapter 29      Configuring Twice NAT

Examples

The following example shows the use of static interface NAT with port translation. Hosts on the outside access an FTP server on the inside by connecting to the outside interface IP address with destination port 65000 through 65004. The traffic is untranslated to the internal FTP server at 192.168.10.100:6500 through :65004. Note that you specify the source port range in the service object (and not the destination port) because you want to translate the source address and port as identified in the command; the destination port is “any.” Because static NAT is bidirectional, “source” and “destination” refers primarily to the command keywords; the actual source and destination address and port in a packet depends on...
which host sent the packet. In this example, connections are originated from outside to inside, so the “source” address and port of the FTP server is actually the destination address and port in the originating packet.

```
hostname(config)# object service FTP_PASV_PORT_RANGE
hostname(config-service-object)# service tcp source range 65000 65004

hostname(config)# object network HOST_FTP_SERVER
hostname(config-network-object)# host 192.168.10.100

hostname(config)# nat (inside,outside) source static HOST_FTP_SERVER interface service FTP_PASV_PORT_RANGE FTP_PASV_PORT_RANGE
```

## Configuring Identity NAT

This section describes how to configure an identity NAT rule using twice NAT. For more information about identity NAT, see the “Identity NAT” section on page 27-11.

### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1**  
Network object:  
object network obj_name  
{host ip_address | subnet  
subnet_address netmask | range  
ip_address_1 ip_address_2}  
Network object group:  
object-group network grp_name  
{network-object {object net_obj_name |  
subnet_address netmask |  
host ip_address} |  
group-object grp_obj_name} | Configure the real source addresses.  
You can configure either a network object or a network object group. For more information, see the “Configuring Objects” section on page 12-3.  
These are the addresses on which you want to perform identity NAT. If you want to perform identity NAT for all addresses, you can skip this step and instead use the keywords any any. |

**Example:**

```
hostname(config)# object network MyInsNet
hostname(config-network-object)# subnet 10.1.1.0 255.255.255.0
```
### Step 2  
(Optional)

**Network object:**

```
object network obj_name  
  (host ip_address | subnet subnet_address netmask | range ip_address_1 ip_address_2)
```

**Network object group:**

```
object-group network grp_name  
  (network-object (object net_obj_name | subnet_address netmask | host ip_address) |  
   group-object grp_obj_name)
```

**Example:**

```
hostname(config)# object network Server1  
hostname(config-network-object)# host 209.165.201.8
```

### Step 3  
(Optional)

**Network object:**

```
object network obj_name  
  (host ip_address | subnet subnet_address netmask | range ip_address_1 ip_address_2)
```

**Network object group:**

```
object-group network grp_name  
  (network-object (object net_obj_name | subnet_address netmask | host ip_address) |  
   group-object grp_obj_name)
```

**Example:**

```
hostname(config)# object network Server1_mapped  
hostname(config-network-object)# host 10.1.1.67
```
### Command

**object service** *obj_name*

- **service** *(tcp | udp)* *(source operator port)* *(destination operator port)*

---

**Example:**

```bash
hostname(config)# object service
REAL_SRC_SVC
hostname(config-service-object)# service
tcp source eq 80

hostname(config)# object service
MAPPED_SRC_SVC
hostname(config-service-object)# service
tcp source eq 8080
```

---

### Purpose

Configure service objects for:

- **Source or destination real port**
- **Source or destination mapped port**

A service object can contain both a source and destination port; however, you should specify *either* the source or the destination port for both service objects. You should only specify *both* the source and destination ports if your application uses a fixed source port (such as some DNS servers); but fixed source ports are rare. NAT only supports TCP or UDP. When translating a port, be sure the protocols in the real and mapped service objects are identical (both TCP or both UDP). For identity NAT, you can use the same service object for both the real and mapped ports. The “not equal” (*neq*) operator is not supported.

For example, if you want to translate the port for the source host, then configure the source service.
### Command

**Step 5**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| `nat (real_ifc,mapped_ifc)`  
  `[line]`  
  `[after-object line]`  
  `source static {nw_obj nw_obj | any any}`  
  `[destination static {mapped_obj | interface} real_obj]`  
  `[service real_srcMapped_dest_svc_obj mapped_src_real_dest_svc_obj]`  
  `[no-proxy-arp] [route-lookup] [inactive] [description desc]` | Configures **identity NAT**. See the following guidelines:  
  - **Interfaces**—(Required for transparent mode) Specify the real and mapped interfaces. Be sure to include the parentheses in your command. In routed mode, if you do not specify the real and mapped interfaces, all interfaces are used; you can also specify the keyword **any** for one or both of the interfaces.  
  - **Section and Line**—(Optional) By default, the NAT rule is added to the end of section 1 of the NAT table. See the “NAT Rule Order” section on page 27-20 for more information about sections. If you want to add the rule into section 3 instead (after the network object NAT rules), then use the **after-auto** keyword. You can insert a rule anywhere in the applicable section using the **line** argument.  
  - **Source addresses**—Specify a network object, group, or the **any** keyword for both the real and mapped addresses (see **Step 1**).  
  - **Destination addresses** (Optional):  
    - **Mapped**—Specify a network object or group, or for static interface NAT with port translation only, specify the **interface** keyword (routed mode only) (see **Step 3**). If you specify **interface**, be sure to also configure the **service** keyword (in this case, the service objects should include only the destination port). For this option, you must configure a specific interface for the **real_ifc**. See the “Static Interface NAT with Port Translation” section on page 27-5 for more information.  
    - **Real**—Specify a network object or group (see **Step 2**). For identity NAT, simply use the same object or group for both the real and mapped addresses.  
  - **Port**—(Optional) Specify the **service** keyword along with the real and mapped service objects (see **Step 4**). For source port translation, the objects must specify the source service. The order of the service objects in the command for source port translation is **service real_obj mapped_obj**. For destination port translation, the objects must specify the destination service. The order of the service objects for destination port translation is **service mappedObj real_obj**. In the rare case where you specify both the source and destination ports in the object, the first service object contains the real source port/mapped destination port; the second service object contains the mapped source port/real destination port. For identity port translation, simply use the same service object for both the real and mapped ports (source and/or destination ports, depending on your configuration).  

Example:

```
hostname(config)# nat (inside,outside)  
source static MyInsNet MyInsNet  
destination static Server1 Server1
```
Chapter 29      Configuring Twice NAT

Monitoring Twice NAT

To monitor twice NAT, enter one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show nat</td>
<td>Shows NAT statistics, including hits for each NAT rule.</td>
</tr>
<tr>
<td>show nat pool</td>
<td>Shows NAT pool statistics, including the addresses and ports allocated, and how many times they were allocated.</td>
</tr>
<tr>
<td>show xlate</td>
<td>Shows current NAT session information.</td>
</tr>
</tbody>
</table>

Configuration Examples for Twice NAT

This section includes the following configuration examples:

- Different Translation Depending on the Destination (Dynamic PAT), page 29-22
- Different Translation Depending on the Destination Address and Port (Dynamic PAT), page 29-24

Different Translation Depending on the Destination (Dynamic PAT)

Figure 29-1 shows a host on the 10.1.2.0/24 network accessing two different servers. When the host accesses the server at 209.165.201.11, the real address is translated to 209.165.202.129:port. When the host accesses the server at 209.165.200.225, the real address is translated to 209.165.202.130:port.
Figure 29-1 Twice NAT with Different Destination Addresses

Step 1  Add a network object for the inside network:

```conf
hostname(config)# object network myInsideNetwork
hostname(config-network-object)# subnet 10.1.2.0 255.255.255.0
```

Step 2  Add a network object for the DMZ network 1:

```conf
hostname(config)# object network DMZnetwork1
hostname(config-network-object)# subnet 209.165.201.0 255.255.255.224
```

Step 3  Add a network object for the PAT address:

```conf
hostname(config)# object network PATaddress1
hostname(config-network-object)# host 209.165.202.129
```

Step 4  Configure the first twice NAT rule:

```conf
hostname(config)# nat (inside,dmz) source dynamic myInsideNetwork PATaddress1 destination static DMZnetwork1 DMZnetwork1
```

Because you do not want to translate the destination address, you need to configure identity NAT for it by specifying the same address for the real and mapped destination addresses.

By default, the NAT rule is added to the end of section 1 of the NAT table. See the “Configuring Dynamic PAT (Hide)” section on page 29-8 for more information about specifying the section and line number for the NAT rule.

Step 5  Add a network object for the DMZ network 2:

```conf
hostname(config)# object network DMZnetwork2
hostname(config-network-object)# subnet 209.165.200.224 255.255.255.224
```

Step 6  Add a network object for the PAT address:

```conf
hostname(config)# object network PATaddress2
```
Step 7 Configure the second twice NAT rule:

```
hostname(config)# nat (inside,dmz) source dynamic myInsideNetwork PATaddress2 destination static DMZnetwork2 DMZnetwork2
```

Different Translation Depending on the Destination Address and Port (Dynamic PAT)

Figure 29-2 shows the use of source and destination ports. The host on the 10.1.2.0/24 network accesses a single host for both web services and Telnet services. When the host accesses the server for Telnet services, the real address is translated to 209.165.202.129:port. When the host accesses the same server for web services, the real address is translated to 209.165.202.130:port.

**Figure 29-2  Twice NAT with Different Destination Ports**

<table>
<thead>
<tr>
<th>Web and Telnet server:</th>
<th>Web Packet Dest. Address:</th>
<th>Telnet Packet Dest. Address:</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.1.2.0/24</td>
<td>10.1.2.27:80</td>
<td>10.1.2.27:23</td>
</tr>
</tbody>
</table>

Step 1 Add a network object for the inside network:

```
hostname(config)# object network myInsideNetwork
hostname(config-network-object)# subnet 10.1.2.0 255.255.255.0
```

Step 2 Add a network object for the Telnet/Web server:

```
hostname(config)# object network TelnetWebServer
```

Step 3 Add a network object for the PAT address when using Telnet:

```
hostname(config)# object network PATaddress1
```
hostname(config-network-object)# host 209.165.202.129

Step 4 Add a service object for Telnet:
hostname(config)# object service TelnetObj
hostname(config-network-object)# service tcp destination eq telnet

Step 5 Configure the first twice NAT rule:
hostname(config)# nat (inside,outside) source dynamic myInsideNetwork PATaddress1
destination static TelnetWebServer TelnetWebServer service TelnetObj TelnetObj

Because you do not want to translate the destination address or port, you need to configure identity NAT for them by specifying the same address for the real and mapped destination addresses, and the same port for the real and mapped service.

By default, the NAT rule is added to the end of section 1 of the NAT table. See the “Configuring Dynamic PAT (Hide)” section on page 29-8 for more information about specifying the section and line number for the NAT rule.

Step 6 Add a network object for the PAT address when using HTTP:
hostname(config)# object network PATaddress2
hostname(config-network-object)# host 209.165.202.130

Step 7 Add a service object for HTTP:
hostname(config)# object service HTTPObj
hostname(config-network-object)# service tcp destination eq http

Step 8 Configure the second twice NAT rule:
hostname(config)# nat (inside,outside) source dynamic myInsideNetwork PATaddress2
destination static TelnetWebServer TelnetWebServer service HTTPObj HTTPObj
## Feature History for Twice NAT

Table 29-1 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Twice NAT</td>
<td>8.3(1)</td>
<td>Twice NAT lets you identify both the source and destination address in a single rule.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>We modified or introduced the following commands: <code>nat</code>, <code>show nat</code>, <code>show xlate</code>, <code>show nat pool</code>.</td>
</tr>
<tr>
<td>Identity NAT configurable proxy ARP and route</td>
<td>8.4(2)/8.5(1)</td>
<td>In earlier releases for identity NAT, proxy ARP was disabled, and a route lookup was always used to determine the egress interface. You could not configure these settings. In 8.4(2) and later, the default behavior for identity NAT was changed to match the behavior of other static NAT configurations: proxy ARP is enabled, and the NAT configuration determines the egress interface (if specified) by default. You can leave these settings as is, or you can enable or disable them discretely. Note that you can now also disable proxy ARP for regular static NAT. For pre-8.3 configurations, the migration of NAT exempt rules (the <code>nat 0 access-list</code> command) to 8.4(2) and later now includes the following keywords to disable proxy ARP and to use a route lookup: <code>no-proxy-arp</code> and <code>route-lookup</code>. The <code>unidirectional</code> keyword that was used for migrating to 8.3(2) and 8.4(1) is no longer used for migration. When upgrading to 8.4(2) from 8.3(1), 8.3(2), and 8.4(1), all identity NAT configurations will now include the <code>no-proxy-arp</code> and <code>route-lookup</code> keywords, to maintain existing functionality. The <code>unidirectional</code> keyword is removed. We modified the following commands: <code>nat source static [no-proxy-arp] [route-lookup]</code>.</td>
</tr>
<tr>
<td>lookup</td>
<td></td>
<td>You can now specify a pool of PAT addresses instead of a single address. You can also optionally enable round-robin assignment of PAT addresses instead of first using all ports on a PAT address before using the next address in the pool. These features help prevent a large number of connections from a single PAT address from appearing to be part of a DoS attack and makes configuration of large numbers of PAT addresses easy. We modified the following commands: <code>nat source dynamic [pat-pool mapped_object [round-robin]]</code>.</td>
</tr>
</tbody>
</table>
PART 8

Configuring Service Policies Using the Modular Policy Framework
Configuring a Service Policy Using the Modular Policy Framework

Service policies using Modular Policy Framework provide a consistent and flexible way to configure ASASM features. For example, you can use a service policy to create a timeout configuration that is specific to a particular TCP application, as opposed to one that applies to all TCP applications. A service policy consists of multiple actions applied to an interface or applied globally.

This chapter includes the following sections:

- Information About Service Policies, page 30-1
- Licensing Requirements for Service Policies, page 30-6
- Guidelines and Limitations, page 30-6
- Default Settings, page 30-7
- Task Flows for Configuring Service Policies, page 30-9
- Identifying Traffic (Layer 3/4 Class Maps), page 30-12
- Defining Actions (Layer 3/4 Policy Map), page 30-15
- Applying Actions to an Interface (Service Policy), page 30-17
- Monitoring Modular Policy Framework, page 30-18
- Configuration Examples for Modular Policy Framework, page 30-18
- Feature History for Service Policies, page 30-21

Information About Service Policies

This section describes how service policies work and includes the following topics:

- Supported Features for Through Traffic, page 30-2
- Supported Features for Management Traffic, page 30-2
- Feature Directionality, page 30-2
- Feature Matching Within a Service Policy, page 30-3
- Order in Which Multiple Feature Actions are Applied, page 30-4
- Incompatibility of Certain Feature Actions, page 30-4
- Feature Matching for Multiple Service Policies, page 30-5
Supported Features for Through Traffic

Table 30-1 lists the features supported by Modular Policy Framework.

Table 30-1     Modular Policy Framework

<table>
<thead>
<tr>
<th>Feature</th>
<th>See:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application inspection (multiple types)</td>
<td>• Chapter 39, “Getting Started with Application Layer Protocol Inspection.”</td>
</tr>
<tr>
<td></td>
<td>• Chapter 40, “Configuring Inspection of Basic Internet Protocols.”</td>
</tr>
<tr>
<td></td>
<td>• Chapter 41, “Configuring Inspection for Voice and Video Protocols.”</td>
</tr>
<tr>
<td></td>
<td>• Chapter 42, “Configuring Inspection of Database and Directory Protocols.”</td>
</tr>
<tr>
<td></td>
<td>• Chapter 43, “Configuring Inspection for Management Application Protocols.”</td>
</tr>
<tr>
<td>QoS input and output policing</td>
<td>Chapter 45, “Configuring QoS.”</td>
</tr>
<tr>
<td>TCP and UDP connection limits and timeouts, and TCP sequence number randomization</td>
<td>Chapter 44, “Configuring Connection Settings.”</td>
</tr>
<tr>
<td>TCP normalization</td>
<td>Chapter 44, “Configuring Connection Settings.”</td>
</tr>
<tr>
<td>TCP state bypass</td>
<td>Chapter 44, “Configuring Connection Settings.”</td>
</tr>
</tbody>
</table>

Supported Features for Management Traffic

Modular Policy Framework supports the following features for management traffic:

• Application inspection for RADIUS accounting traffic—See Chapter 43, “Configuring Inspection for Management Application Protocols.”

• Connection limits—See Chapter 44, “Configuring Connection Settings.”

Feature Directionality

Actions are applied to traffic bidirectionally or unidirectionally depending on the feature. For features that are applied bidirectionally, all traffic that enters or exits the interface to which you apply the policy map is affected if the traffic matches the class map for both directions.

Note

When you use a global policy, all features are unidirectional; features that are normally bidirectional when applied to a single interface only apply to the ingress of each interface when applied globally. Because the policy is applied to all interfaces, the policy will be applied in both directions so bidirectionality in this case is redundant.
For features that are applied unidirectionally, for example QoS priority queue, only traffic that enters (or exits, depending on the feature) the interface to which you apply the policy map is affected. See Table 30-2 for the directionality of each feature.

### Table 30-2  Feature Directionality

<table>
<thead>
<tr>
<th>Feature</th>
<th>Single Interface Direction</th>
<th>Global Direction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application inspection (multiple types)</td>
<td>Bidirectional</td>
<td>Ingress</td>
</tr>
<tr>
<td>ASA CSC</td>
<td>Bidirectional</td>
<td>Ingress</td>
</tr>
<tr>
<td>ASA CX</td>
<td>Bidirectional</td>
<td>Ingress</td>
</tr>
<tr>
<td>ASA CX authentication proxy</td>
<td>Ingress</td>
<td>Ingress</td>
</tr>
<tr>
<td>ASA IPS</td>
<td>Bidirectional</td>
<td>Ingress</td>
</tr>
<tr>
<td>NetFlow Secure Event Logging filtering</td>
<td>N/A</td>
<td>Ingress</td>
</tr>
<tr>
<td>QoS input policing</td>
<td>Ingress</td>
<td>Ingress</td>
</tr>
<tr>
<td>QoS output policing</td>
<td>Egress</td>
<td>Egress</td>
</tr>
<tr>
<td>QoS standard priority queue</td>
<td>Egress</td>
<td>Egress</td>
</tr>
<tr>
<td>QoS traffic shaping, hierarchical priority queue</td>
<td>Egress</td>
<td>Egress</td>
</tr>
<tr>
<td>TCP and UDP connection limits and timeouts, and TCP sequence number randomization</td>
<td>Bidirectional</td>
<td>Ingress</td>
</tr>
<tr>
<td>TCP normalization</td>
<td>Bidirectional</td>
<td>Ingress</td>
</tr>
<tr>
<td>TCP state bypass</td>
<td>Bidirectional</td>
<td>Ingress</td>
</tr>
</tbody>
</table>

### Feature Matching Within a Service Policy

See the following information for how a packet matches class maps in a policy map for a given interface:

1. A packet can match only one class map in the policy map for each feature type.
2. When the packet matches a class map for a feature type, the ASASM does not attempt to match it to any subsequent class maps for that feature type.
3. If the packet matches a subsequent class map for a different feature type, however, then the ASASM also applies the actions for the subsequent class map, if supported. See the “Incompatibility of Certain Feature Actions” section on page 30-4 for more information about unsupported combinations.

**Note**  Application inspection includes multiple inspection types, and most are mutually exclusive. For inspections that can be combined, each inspection is considered to be a separate feature.

For example, if a packet matches a class map for connection limits, and also matches a class map for an application inspection, then both actions are applied.

If a packet matches a class map for HTTP inspection, but also matches another class map that includes HTTP inspection, then the second class map actions are not applied.

If a packet matches a class map for HTTP inspection, but also matches another class map that includes FTP inspection, then the second class map actions are not applied because HTTP and FTP inspections cannot be combined.
If a packet matches a class map for HTTP inspection, but also matches another class map that includes IPv6 inspection, then both actions are applied because the IPv6 inspection can be combined with any other type of inspection.

**Order in Which Multiple Feature Actions are Applied**

The order in which different types of actions in a policy map are performed is independent of the order in which the actions appear in the policy map.

*Note*

NetFlow Secure Event Logging filtering is order-independent.

Actions are performed in the following order:

1. QoS input policing
2. TCP normalization, TCP and UDP connection limits and timeouts, TCP sequence number randomization, and TCP state bypass.

*Note*

When the ASASM performs a proxy service (such as AAA or CSC) or it modifies the TCP payload (such as FTP inspection), the TCP normalizer acts in dual mode, where it is applied before and after the proxy or payload modifying service.

3. ASA CSC
4. Application inspections that can be combined with other inspections:
   a. IPv6
   b. IP options
   c. WAAS
5. Application inspections that cannot be combined with other inspections. The remaining application inspections cannot be combined with other inspections. See the “Incompatibility of Certain Feature Actions” section on page 30-4 for more information.
6. ASA IPS
7. ASA CX
8. QoS output policing
9. QoS standard priority queue
10. QoS traffic shaping, hierarchical priority queue

**Incompatibility of Certain Feature Actions**

Some features are not compatible with each other for the same traffic. The following list may not include all incompatibilities; for information about compatibility of each feature, see the chapter or section for your feature:

- You cannot configure QoS priority queueing and QoS policing for the same set of traffic.
Most inspections should not be combined with another inspection, so the ASASM only applies one inspection if you configure multiple inspections for the same traffic. The only exceptions are listed in the “Order in Which Multiple Feature Actions are Applied” section on page 30-4.

You cannot configure traffic to be sent to multiple modules, such as the ASA CX and ASA IPS.

HTTP inspection is not compatible with the ASA CX.

Note

The **match default-inspection-traffic** command, which is used in the default global policy, is a special CLI shortcut to match the default ports for all inspections. When used in a policy map, this class map ensures that the correct inspection is applied to each packet, based on the destination port of the traffic. For example, when UDP traffic for port 69 reaches the ASASM, then the ASASM applies the TFTP inspection; when TCP traffic for port 21 arrives, then the ASASM applies the FTP inspection. So in this case only, you can configure multiple inspections for the same class map. Normally, the ASASM does not use the port number to determine which inspection to apply, thus giving you the flexibility to apply inspections to non-standard ports, for example.

An example of a misconfiguration is if you configure multiple inspections in the same policy map and do not use the default-inspection-traffic shortcut. In **Example 30-1**, traffic destined to port 21 is mistakenly configured for both FTP and HTTP inspection. In **Example 30-2**, traffic destined to port 80 is mistakenly configured for both FTP and HTTP inspection. In both cases of misconfiguration examples, only the FTP inspection is applied, because FTP comes before HTTP in the order of inspections applied.

**Example 30-1  Misconfiguration for FTP packets: HTTP Inspection Also Configured**

```plaintext
class-map ftp
  match port tcp eq 21
class-map http
  match port tcp eq 21 [it should be 80]
policy-map test
  class ftp
    inspect ftp
  class http
    inspect http
```

**Example 30-2  Misconfiguration for HTTP packets: FTP Inspection Also Configured**

```plaintext
class-map ftp
  match port tcp eq 80 [it should be 21]
class-map http
  match port tcp eq 80
policy-map test
  class http
    inspect http
  class ftp
    inspect ftp
```

**Feature Matching for Multiple Service Policies**

For TCP and UDP traffic (and ICMP when you enable stateful ICMP inspection), service policies operate on traffic flows, and not just individual packets. If traffic is part of an existing connection that matches a feature in a policy on one interface, that traffic flow cannot also match the same feature in a policy on another interface; only the first policy is used.
For example, if HTTP traffic matches a policy on the inside interface to inspect HTTP traffic, and you have a separate policy on the outside interface for HTTP inspection, then that traffic is not also inspected on the egress of the outside interface. Similarly, the return traffic for that connection will not be inspected by the ingress policy of the outside interface, nor by the egress policy of the inside interface.

For traffic that is not treated as a flow, for example ICMP when you do not enable stateful ICMP inspection, returning traffic can match a different policy map on the returning interface. For example, if you configure IPS on the inside and outside interfaces, but the inside policy uses virtual sensor 1 while the outside policy uses virtual sensor 2, then a non-stateful Ping will match virtual sensor 1 outbound, but will match virtual sensor 2 inbound.

Licensing Requirements for Service Policies

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

Specific features may have separate license requirements. See the feature chapter for more information.

Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

Context Mode Guidelines
Supported in single and multiple context mode.

Firewall Mode Guidelines
Supported in routed and transparent firewall mode.

IPv6 Guidelines
Supports IPv6 for the following features:
- Application inspection for FTP, HTTP, ICMP, SIP, SMTP and IPsec-pass-thru, and IPv6.
- ASA IPS
- ASA CX
- NetFlow Secure Event Logging filtering
- TCP and UDP connection limits and timeouts, TCP sequence number randomization
- TCP normalization
- TCP state bypass

Class Map Guidelines
The maximum number of class maps of all types is 255 in single mode or per context in multiple mode. Class maps include the following types:
- Layer 3/4 class maps (for through traffic and management traffic).
- Inspection class maps
Regular expression class maps

match commands used directly underneath an inspection policy map

This limit also includes default class maps of all types, limiting user-configured class maps to approximately 235. See the “Default Class Maps” section on page 30-8.

Policy Map Guidelines

See the following guidelines for using policy maps:

- You can only assign one policy map per interface. (However you can create up to 64 policy maps in the configuration.)
- You can apply the same policy map to multiple interfaces.
- You can identify up to 63 Layer 3/4 class maps in a Layer 3/4 policy map.
- For each class map, you can assign multiple actions from one or more feature types, if supported. See the “Incompatibility of Certain Feature Actions” section on page 30-4.

Service Policy Guidelines

- Interface service policies take precedence over the global service policy for a given feature. For example, if you have a global policy with FTP inspection, and an interface policy with TCP normalization, then both FTP inspection and TCP normalization are applied to the interface. However, if you have a global policy with FTP inspection, and an interface policy with FTP inspection, then only the interface policy FTP inspection is applied to that interface.
- You can only apply one global policy. For example, you cannot create a global policy that includes feature set 1, and a separate global policy that includes feature set 2. All features must be included in a single policy.

Default Settings

The following topics describe the default settings for Modular Policy Framework:

- Default Configuration, page 30-7
- Default Class Maps, page 30-8

Default Configuration

By default, the configuration includes a policy that matches all default application inspection traffic and applies certain inspections to the traffic on all interfaces (a global policy). Not all inspections are enabled by default. You can only apply one global policy, so if you want to alter the global policy, you need to either edit the default policy or disable it and apply a new one. (An interface policy overrides the global policy for a particular feature.)

The default policy includes the following application inspections:

- DNS inspection for the maximum message length of 512 bytes
- FTP
- H323 (H225)
- H323 (RAS)
- RSH
Default Settings

- RTSP
- ESMTP
- SQLnet
- Skinny (SCCP)
- SunRPC
- XDMCP
- SIP
- NetBIOS
- TFTP
- IP Options

The default policy configuration includes the following commands:

```plaintext
class-map inspection_default
  match default-inspection-traffic
policy-map type inspect dns preset_dns_map
  parameters
    message-length maximum 512
policy-map global_policy
  class inspection_default
    inspect dns preset_dns_map
    inspect ftp
    inspect h323 h225
    inspect h323 ras
    inspect rsh
    inspect rtsp
    inspect esmtp
    inspect sqlnet
    inspect skinny
    inspect sunrpc
    inspect xdmcp
    inspect sip
    inspect netbios
    inspect tftp
    inspect ip-options
service-policy global_policy global
```

**Note:** See the “Incompatibility of Certain Feature Actions” section on page 30-4 for more information about the special `match default-inspection-traffic` command used in the default class map.

Default Class Maps

The configuration includes a default Layer 3/4 class map that the ASASM uses in the default global policy called default-inspection-traffic; it matches the default inspection traffic. This class, which is used in the default global policy, is a special shortcut to match the default ports for all inspections. When used in a policy, this class ensures that the correct inspection is applied to each packet, based on the destination port of the traffic. For example, when UDP traffic for port 69 reaches the ASASM, then the ASASM applies the TFTP inspection; when TCP traffic for port 21 arrives, then the ASASM applies the FTP inspection. So in this case only, you can configure multiple inspections for the same class map. Normally, the ASASM does not use the port number to determine which inspection to apply, thus giving you the flexibility to apply inspections to non-standard ports, for example.

```plaintext
class-map inspection_default
```
Another class map that exists in the default configuration is called class-default, and it matches all traffic. This class map appears at the end of all Layer 3/4 policy maps and essentially tells the ASASM to not perform any actions on all other traffic. You can use the class-default class if desired, rather than making your own match any class map. In fact, some features are only available for class-default, such as QoS traffic shaping.

```
class-map class-default
  match any
```

## Task Flows for Configuring Service Policies

This section includes the following topics:

- Task Flow for Using the Modular Policy Framework, page 30-9

## Task Flow for Using the Modular Policy Framework

To configure Modular Policy Framework, perform the following steps:

### Step 1
Identify the traffic—Identify the traffic on which you want to perform Modular Policy Framework actions by creating Layer 3/4 class maps.

For example, you might want to perform actions on all traffic that passes through the ASASM; or you might only want to perform certain actions on traffic from 10.1.1.0/24 to any destination address.

See the “Identifying Traffic (Layer 3/4 Class Maps)” section on page 30-12.

### Step 2
Perform additional actions on some inspection traffic—If one of the actions you want to perform is application inspection, and you want to perform additional actions on some inspection traffic, then create an inspection policy map. The inspection policy map identifies the traffic and specifies what to do with it.

For example, you might want to drop all HTTP requests with a body length greater than 1000 bytes.
You can create a self-contained inspection policy map that identifies the traffic directly with match commands, or you can create an inspection class map for reuse or for more complicated matching. See the “Defining Actions in an Inspection Policy Map” section on page 31-2 and the “Identifying Traffic in an Inspection Class Map” section on page 31-6.

**Step 3**  Create a regular expression—If you want to match text with a regular expression within inspected packets, you can create a regular expression or a group of regular expressions (a regular expression class map). Then, when you define the traffic to match for the inspection policy map, you can call on an existing regular expression.

For example, you might want to drop all HTTP requests with a URL including the text “example.com.”

See the “Creating a Regular Expression” section on page 12-12 and the “Creating a Regular Expression Class Map” section on page 12-15.

**Step 4**  Define the actions you want to perform and determine on which interfaces you want to apply the policy map—Define the actions you want to perform on each Layer 3/4 class map by creating a Layer 3/4 policy map. Then, determine on which interfaces you want to apply the policy map using a service policy.
See the “Defining Actions (Layer 3/4 Policy Map)” section on page 30-15 and the “Applying Actions to an Interface (Service Policy)” section on page 30-17.

**Task Flow for Configuring Hierarchical Policy Maps for QoS Traffic Shaping**

If you enable QoS traffic shaping for a class map, then you can optionally enable priority queueing for a subset of shaped traffic. To do so, you need to create a policy map for the priority queueing, and then within the traffic shaping policy map, you can call the priority class map. Only the traffic shaping class map is applied to an interface.

See Chapter 45, “Information About QoS,” for more information about this feature.

Hierarchical policy maps are only supported for traffic shaping and priority queueing.

To implement a hierarchical policy map, perform the following steps:

**Step 1** Identify the prioritized traffic according to the “Identifying Traffic (Layer 3/4 Class Maps)” section on page 30-12.
You can create multiple class maps to be used in the hierarchical policy map.

**Step 2** Create a policy map according to the “Defining Actions (Layer 3/4 Policy Map)” section on page 30-15, and identify the sole action for each class map as **priority**.

**Step 3** Create a separate policy map according to the “Defining Actions (Layer 3/4 Policy Map)” section on page 30-15, and identify the **shape** action for the **class-default** class map.
Traffic shaping can only be applied to the class-default class map.

**Step 4** For the same class map, identify the priority policy map that you created in Step 2 using the `service-policy priority_policy_map` command.

**Step 5** Apply the shaping policy map to the interface according to “Applying Actions to an Interface (Service Policy)” section on page 30-17.

---

### Identifying Traffic (Layer 3/4 Class Maps)

A Layer 3/4 class map identifies Layer 3 and 4 traffic to which you want to apply actions. You can create multiple Layer 3/4 class maps for each Layer 3/4 policy map.

This section includes the following topics:
- Creating a Layer 3/4 Class Map for Through Traffic, page 30-12
- Creating a Layer 3/4 Class Map for Management Traffic, page 30-14

### Creating a Layer 3/4 Class Map for Through Traffic

A Layer 3/4 class map matches traffic based on protocols, ports, IP addresses and other Layer 3 or 4 attributes.

**Detailed Steps**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> class-map class_map_name</td>
<td>Creates a Layer 3/4 class map, where <code>class_map_name</code> is a string up to 40 characters in length. The name “class-default” is reserved. All types of class maps use the same name space, so you cannot reuse a name already used by another type of class map. The CLI enters class-map configuration mode.</td>
</tr>
<tr>
<td>Example: hostname(config)# class-map all_udp</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> (Optional) description string</td>
<td>Adds a description to the class map.</td>
</tr>
<tr>
<td>Example: hostname(config-cmap)# description All UDP traffic</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> Match traffic using one of the following:</td>
<td>Unless otherwise specified, you can include only one <code>match</code> command in the class map.</td>
</tr>
<tr>
<td>match any</td>
<td>Matches all traffic.</td>
</tr>
<tr>
<td>Example: hostname(config-cmap)# match any</td>
<td></td>
</tr>
</tbody>
</table>
### Identifying Traffic (Layer 3/4 Class Maps)

#### match access-list access_list_name

- **Purpose:** Matches traffic specified by an extended access list. If the ASASM is operating in transparent firewall mode, you can use an EtherType access list.

  **Example:**
  ```
  hostname(config-cmap)# match access-list udp
  ```

#### match port (tcp | udp) {eq port_num | range port_num port_num}

- **Purpose:** Matches TCP or UDP destination ports, either a single port or a contiguous range of ports.

  **Tip:** For applications that use multiple, non-contiguous ports, use the `match access-list` command and define an ACE to match each port.

  **Example:**
  ```
  hostname(config-cmap)# match tcp eq 80
  ```

#### match default-inspection-traffic

- **Purpose:** Matches default traffic for inspection: the default TCP and UDP ports used by all applications that the ASASM can inspect. This command, which is used in the default global policy, is a special CLI shortcut that when used in a policy map, ensures that the correct inspection is applied to each packet, based on the destination port of the traffic. For example, when UDP traffic for port 69 reaches the ASASM, then the ASASM applies the TFTP inspection; when TCP traffic for port 21 arrives, then the ASASM applies the FTP inspection. So in this case only, you can configure multiple inspections for the same class map (with the exception of WAAS inspection, which can be configured with other inspections. See the “Incompatibility of Certain Feature Actions” on page 30-4 for more information about combining actions). Normally, the ASASM does not use the port number to determine the inspection applied, thus giving you the flexibility to apply inspections to non-standard ports, for example.

  **Example:**
  ```
  hostname(config-cmap)# match default-inspection-traffic
  ```

  **See:** The “Default Settings” section on page 39-4 for a list of default ports. Not all applications whose ports are included in the `match default-inspection-traffic` command are enabled by default in the policy map.

  You can specify a `match access-list` command along with the `match default-inspection-traffic` command to narrow the matched traffic. Because the `match default-inspection-traffic` command specifies the ports and protocols to match, any ports and protocols in the access list are ignored.

  **Tip:** We suggest that you only inspect traffic on ports on which you expect application traffic; if you inspect all traffic, for example using `match any`, the ASASM performance can be impacted.

  **Example:**
  ```
  hostname(config-cmap)# match dscp af43 cs1 ef
  ```

#### match dscp value1 [value2] [...] [value8]

- **Purpose:** Matches DSCP value in an IP header, up to eight DSCP values.
Chapter 30 Configuring a Service Policy Using the Modular Policy Framework

Identifying Traffic (Layer 3/4 Class Maps)

### Command

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>match precedence</strong> <code>value1</code> <code>[value2]</code> <code>[value3]</code> <code>[value4]</code></td>
<td>Matches up to four precedence values, represented by the TOS byte in the IP header, where <code>value1</code> through <code>value4</code> can be 0 to 7, corresponding to the possible precedences.</td>
</tr>
<tr>
<td><strong>match rtp</strong> <code>starting_port range</code></td>
<td>Matches RTP traffic, where the <code>starting_port</code> specifies an even-numbered UDP destination port between 2000 and 65534. The <code>range</code> specifies the number of additional UDP ports to match above the <code>starting_port</code>, between 0 and 16383.</td>
</tr>
<tr>
<td><strong>match tunnel-group</strong> <code>name</code> (Optional)</td>
<td>Matches VPN tunnel group traffic to which you want to apply QoS. You can also specify one other <code>match</code> command to refine the traffic match. You can specify any of the preceding commands, except for the <code>match any</code>, <code>match access-list</code>, or <code>match default-inspection-traffic</code> commands. Or you can also enter the <code>match flow ip destination-address</code> command to match flows in the tunnel group going to each IP address.</td>
</tr>
<tr>
<td><strong>match flow ip destination-address</strong></td>
<td></td>
</tr>
</tbody>
</table>

### Examples

The following is an example for the `class-map` command:

```bash
hostname(config)# access-list udp permit udp any any
hostname(config)# access-list tcp permit tcp any any
hostname(config)# access-list host_foo permit ip any 10.1.1.1 255.255.255.255

hostname(config)# class-map all_udp
hostname(config-cmap)# description "This class-map matches all UDP traffic"
hostname(config-cmap)# match access-list udp

hostname(config-cmap)# class-map all_tcp
hostname(config-cmap)# description "This class-map matches all TCP traffic"
hostname(config-cmap)# match access-list tcp

hostname(config-cmap)# class-map all_http
hostname(config-cmap)# description "This class-map matches all HTTP traffic"
hostname(config-cmap)# match port tcp eq http

hostname(config-cmap)# class-map to_server
hostname(config-cmap)# description "This class-map matches all traffic to server 10.1.1.1"
hostname(config-cmap)# match access-list host_foo
```

### Creating a Layer 3/4 Class Map for Management Traffic

For management traffic to the ASASM, you might want to perform actions specific to this kind of traffic. You can specify a management class map that can match an access list or TCP or UDP ports. The types of actions available for a management class map in the policy map are specialized for management traffic. See the “Supported Features for Management Traffic” section on page 30-2.
Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> class-map type management class_map_name</td>
<td>Creates a management class map, where class_map_name is a string up to 40 characters in length. The name “class-default” is reserved. All types of class maps use the same name space, so you cannot reuse a name already used by another type of class map. The CLI enters class-map configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# class-map type management all_mgmt</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> (Optional) description string</td>
<td>Adds a description to the class map.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config-cmap)# description All management traffic</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> Match traffic using one of the following:</td>
<td>Unless otherwise specified, you can include only one match command in the class map.</td>
</tr>
<tr>
<td>match access-list access_list_name</td>
<td>Matches traffic specified by an extended access list. If the ASASM is operating in transparent firewall mode, you can use an EtherType access list.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config-cmap)# match access-list udp</td>
<td></td>
</tr>
<tr>
<td>match port (tcp</td>
<td>udp) {eq port_num</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config-cmap)# match tcp eq 80</td>
<td></td>
</tr>
</tbody>
</table>

**Defining Actions (Layer 3/4 Policy Map)**

This section describes how to associate actions with Layer 3/4 class maps by creating a Layer 3/4 policy map.

**Restrictions**

The maximum number of policy maps is 64, but you can only apply one policy map per interface.
## Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>policy-map policy_map_name</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>(Optional) class class_map_name</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>Specify one or more actions for this class map.</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>Repeat Step 2 and Step 3 for each class map you want to include in this policy map.</td>
</tr>
</tbody>
</table>

## Examples

The following is an example of a `policy-map` command for connection policy. It limits the number of connections allowed to the web server 10.1.1.1:

```
hostname(config)# access-list http-server permit tcp any host 10.1.1.1
hostname(config)# class-map http-server
hostname(config-cmap)# match access-list http-server
hostname(config)# policy-map global-policy
hostname(config-pmap)# description This policy map defines a policy concerning connection to http server.
hostname(config-pmap)# class http-server
hostname(config-pmap-c)# set connection conn-max 256
```

The following example shows how multi-match works in a policy map:

```
hostname(config)# class-map inspection_default
hostname(config-cmap)# match default-inspection-traffic
hostname(config)# class-map http_traffic
hostname(config-cmap)# match port tcp eq 80

hostname(config)# policy-map outside_policy
hostname(config-pmap)# class inspection_default
hostname(config-pmap-c)# inspect http http_map
hostname(config-pmap-c)# inspect sip
hostname(config-pmap-c)# class http_traffic
hostname(config-pmap-c)# set connection timeout idle 0:10:0
```
The following example shows how traffic matches the first available class map, and will not match any subsequent class maps that specify actions in the same feature domain:

hostname(config)# class-map telnet_traffic
hostname(config-cmap)# match port tcp eq 23
hostname(config-cmap)# class-map ftp_traffic
hostname(config-cmap)# match port tcp eq 21
hostname(config-cmap)# class-map tcp_traffic
hostname(config-cmap)# match port tcp range 1 65535
hostname(config-cmap)# class-map udp_traffic
hostname(config-cmap)# match port udp range 0 65535
hostname(config)# policy-map global_policy
hostname(config-pmap)# class telnet_traffic
hostname(config-pmap-c)# set connection timeout idle 0:0:0
hostname(config-pmap-c)# set connection conn-max 100
hostname(config-pmap-c)# class ftp_traffic
hostname(config-pmap-c)# set connection timeout idle 0:5:0
hostname(config-pmap-c)# set connection conn-max 50
hostname(config-pmap-c)# class tcp_traffic
hostname(config-pmap-c)# set connection timeout idle 2:0:0
hostname(config-pmap-c)# set connection conn-max 2000

When a Telnet connection is initiated, it matches class telnet_traffic. Similarly, if an FTP connection is initiated, it matches class ftp_traffic. For any TCP connection other than Telnet and FTP, it will match class tcp_traffic. Even though a Telnet or FTP connection can match class tcp_traffic, the ASASM does not make this match because they previously matched other classes.

Applying Actions to an Interface (Service Policy)

To activate the Layer 3/4 policy map, create a service policy that applies it to one or more interfaces or that applies it globally to all interfaces.

Restrictions

You can only apply one global policy, so if you want to alter the global policy, you need to either edit the default policy or disable it and apply a new one. By default, the configuration includes a global policy that matches all default application inspection traffic and applies inspection to the traffic globally. The default service policy includes the following command:

global_policy global
Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>service-policy policy_map_name interface interface_name</td>
<td>Creates a service policy by associating a policy map with an interface.</td>
</tr>
</tbody>
</table>

Example:

```
hostname(config)# service-policy inbound_policy interface outside
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>service-policy policy_map_name global</td>
<td>Creates a service policy that applies to all interfaces that do not have a specific policy.</td>
</tr>
</tbody>
</table>

Example:

```
hostname(config)# service-policy inbound_policy global
```

Examples

For example, the following command enables the inbound_policy policy map on the outside interface:

```
hostname(config)# service-policy inbound_policy interface outside
```

The following commands disable the default global policy, and enables a new one called new_global_policy on all other ASASM interfaces:

```
hostname(config)# no service-policy global_policy global
hostname(config)# service-policy new_global_policy global
```

Monitoring Modular Policy Framework

To monitor Modular Policy Framework, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show service-policy</td>
<td>Displays the service policy statistics.</td>
</tr>
</tbody>
</table>

Configuration Examples for Modular Policy Framework

This section includes several Modular Policy Framework examples and includes the following topics:

- Applying Inspection and QoS Policing to HTTP Traffic, page 30-19
- Applying Inspection to HTTP Traffic Globally, page 30-19
- Applying Inspection and Connection Limits to HTTP Traffic to Specific Servers, page 30-20
- Applying Inspection to HTTP Traffic with NAT, page 30-21
Applying Inspection and QoS Policing to HTTP Traffic

In this example (see Figure 30-1), any HTTP connection (TCP traffic on port 80) that enters or exits the ASASM through the outside interface is classified for HTTP inspection. Any HTTP traffic that exits the outside interface is classified for policing.

Figure 30-1 HTTP Inspection and QoS Policing

See the following commands for this example:

```
hostname(config)# class-map http_traffic
hostname(config-cmap)# match port tcp eq 80

hostname(config)# policy-map http_traffic_policy
hostname(config-pmap)# class http_traffic
hostname(config-pmap-c)# inspect http
hostname(config-pmap-c)# police output 250000
hostname(config)# service-policy http_traffic_policy interface outside
```

Applying Inspection to HTTP Traffic Globally

In this example (see Figure 30-2), any HTTP connection (TCP traffic on port 80) that enters the ASASM through any interface is classified for HTTP inspection. Because the policy is a global policy, inspection occurs only as the traffic enters each interface.

Figure 30-2 Global HTTP Inspection

See the following commands for this example:

```
hostname(config)# class-map http_traffic
hostname(config-cmap)# match port tcp eq 80
```
hostname(config)# policy-map http_traffic_policy
hostname(config-pmap)# class http_traffic
hostname(config-pmap-c)# inspect http
hostname(config)# service-policy http_traffic_policy global

Applying Inspection and Connection Limits to HTTP Traffic to Specific Servers

In this example (see Figure 30-3), any HTTP connection destined for Server A (TCP traffic on port 80) that enters the ASASM through the outside interface is classified for HTTP inspection and maximum connection limits. Connections initiated from Server A to Host A does not match the access list in the class map, so it is not affected.

Any HTTP connection destined for Server B that enters the ASASM through the inside interface is classified for HTTP inspection. Connections initiated from Server B to Host B does not match the access list in the class map, so it is not affected.

Figure 30-3 HTTP Inspection and Connection Limits to Specific Servers

See the following commands for this example:

hostname(config)# object network obj-192.168.1.2
hostname(config-network-object)# host 192.168.1.2
hostname(config-network-object)# nat (inside,outside) static 209.165.201.1
hostname(config)# object network obj-192.168.1.0
hostname(config-network-object)# subnet 192.168.1.0 255.255.255.0
hostname(config-network-object)# nat (inside,outside) dynamic 209.165.201.2
hostname(config)# access-list serverA extended permit tcp any host 209.165.201.1 eq 80
hostname(config)# access-list ServerB extended permit tcp any host 209.165.200.227 eq 80

hostname(config)# class-map http_serverA
hostname(config-cmap)# match access-list serverA
hostname(config)# class-map http_serverB
hostname(config-cmap)# match access-list serverB

hostname(config)# policy-map policy_serverA
hostname(config-pmap)# class http_serverA
hostname(config-pmap-c)# inspect http
hostname(config-pmap-c)# set connection conn-max 100
hostname(config)# policy-map policy_serverB
hostname(config-pmap)# class http_serverB
hostname(config-pmap-c)# inspect http
hostname(config)# service-policy policy_serverB interface inside
hostname(config)# service-policy policy_serverA interface outside

Applying Inspection to HTTP Traffic with NAT

In this example, the Host on the inside network has two addresses: one is the real IP address 192.168.1.1, and the other is a mapped IP address used on the outside network, 209.165.200.225. Because the policy is applied to the inside interface, where the real address is used, then you must use the real IP address in the access list in the class map. If you applied it to the outside interface, you would use the mapped address.

![Figure 30-4 HTTP Inspection with NAT](image)

See the following commands for this example:

hostname(config)# static (inside,outside) 209.165.200.225 192.168.1.1
hostname(config)# access-list http_client extended permit tcp host 192.168.1.1 any eq 80
hostname(config)# class-map http_client
hostname(config-cmap)# match access-list http_client
hostname(config)# policy-map http_client
hostname(config-pmap)# class http_client
hostname(config-pmap-c)# inspect http
hostname(config)# service-policy http_client interface inside

Feature History for Service Policies

Table 30-3 lists the release history for this feature.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modular Policy Framework</td>
<td>7.0(1)</td>
<td>Modular Policy Framework was introduced.</td>
</tr>
<tr>
<td>Management class map for use with RADIUS accounting traffic</td>
<td>7.2(1)</td>
<td>The management class map was introduced for use with RADIUS accounting traffic. The following commands were introduced: class-map type management, and inspect radius-accounting.</td>
</tr>
</tbody>
</table>
### Feature History for Service Policies (continued)

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inspection policy maps</td>
<td>7.2(1)</td>
<td>The inspection policy map was introduced. The following command was introduced: <strong>class-map type inspect</strong>.</td>
</tr>
<tr>
<td>Regular expressions and policy maps</td>
<td>7.2(1)</td>
<td>Regular expressions and policy maps were introduced to be used under inspection policy maps. The following commands were introduced: <strong>class-map type regex, regex, match regex</strong>.</td>
</tr>
<tr>
<td>Match any for inspection policy maps</td>
<td>8.0(2)</td>
<td>The <strong>match any</strong> keyword was introduced for use with inspection policy maps: traffic can match one or more criteria to match the class map. Formerly, only <strong>match all</strong> was available.</td>
</tr>
<tr>
<td>Maximum connections and embryonic connections for management traffic</td>
<td>8.0(2)</td>
<td>The <strong>set connection</strong> command is now available for a Layer 3/4 management class map, for to-the-security appliance management traffic. Only the <strong>conn-max</strong> and <strong>embryonic-conn-max</strong> keywords are available.</td>
</tr>
</tbody>
</table>
Configuring Special Actions for Application Inspections (Inspection Policy Map)

Modular Policy Framework lets you configure special actions for many application inspections. When you enable an inspection engine in the Layer 3/4 policy map, you can also optionally enable actions as defined in an inspection policy map. When the inspection policy map matches traffic within the Layer 3/4 class map for which you have defined an inspection action, then that subset of traffic will be acted upon as specified (for example, dropped or rate-limited).

This chapter includes the following sections:
- Information About Inspection Policy Maps, page 31-1
- Guidelines and Limitations, page 31-2
- Default Inspection Policy Maps, page 31-2
- Defining Actions in an Inspection Policy Map, page 31-2
- Identifying Traffic in an Inspection Class Map, page 31-6
- Where to Go Next, page 31-7

Information About Inspection Policy Maps

See the “Configuring Application Layer Protocol Inspection” section on page 39-6 for a list of applications that support inspection policy maps.

An inspection policy map consists of one or more of the following elements. The exact options available for an inspection policy map depends on the application.

- Traffic matching command—You can define a traffic matching command directly in the inspection policy map to match application traffic to criteria specific to the application, such as a URL string, for which you then enable actions.
  - Some traffic matching commands can specify regular expressions to match text inside a packet. Be sure to create and test the regular expressions before you configure the policy map, either singly or grouped together in a regular expression class map.
- Inspection class map—(Not available for all applications. See the CLI help for a list of supported applications.) An inspection class map includes traffic matching commands that match application traffic with criteria specific to the application, such as a URL string. You then identify the class map in the policy map and enable actions. The difference between creating a class map and defining the traffic match directly in the inspection policy map is that you can create more complex match criteria and you can reuse class maps.
- Some traffic matching commands can specify regular expressions to match text inside a packet. Be sure to create and test the regular expressions before you configure the policy map, either singly or grouped together in a regular expression class map.

- Parameters—Parameters affect the behavior of the inspection engine.

## Guidelines and Limitations

- HTTP inspection policy maps—If you modify an in-use HTTP inspection policy map (policy-map type inspect http), you must remove and reapply the inspect http map action for the changes to take effect. For example, if you modify the “http-map” inspection policy map, you must remove and read the inspect http http-map command from the layer 3/4 policy:

```
hostname(config)# policy-map test
hostname(config-pmap)# class http0
hostname(config-pmap-c)# no inspect http http-map
hostname(config-pmap-c)# inspect http http-map
```

- All inspection policy maps—If you want to exchange an in-use inspection policy map for a different map name, you must remove the inspect protocol map command, and readd it with the new map. For example:

```
hostname(config)# policy-map test
hostname(config-pmap)# class sip
hostname(config-pmap-c)# no inspect sip sip-map1
hostname(config-pmap-c)# inspect sip sip-map2
```

## Default Inspection Policy Maps

The default inspection policy map configuration includes the following commands, which sets the maximum message length for DNS packets to be 512 bytes:

```
policy-map type inspect dns preset_dns_map
  parameters
    message-length maximum 512
```

**Note**

There are other default inspection policy maps such as policy-map type inspect esmtp _default_esmtp_map. These default policy maps are created implicitly by the command inspect protocol. For example, inspect esmtp implicitly uses the policy map “_default_esmtp_map.” All the default policy maps can be shown by using the show running-config all policy-map command.

## Defining Actions in an Inspection Policy Map

When you enable an inspection engine in the Layer 3/4 policy map, you can also optionally enable actions as defined in an inspection policy map.

**Restrictions**

You can specify multiple class or match commands in the policy map.
If a packet matches multiple different **match** or **class** commands, then the order in which the ASASM applies the actions is determined by internal ASASM rules, and not by the order they are added to the policy map. The internal rules are determined by the application type and the logical progression of parsing a packet, and are not user-configurable. For example for HTTP traffic, parsing a Request Method field precedes parsing the Header Host Length field; an action for the Request Method field occurs before the action for the Header Host Length field. For example, the following match commands can be entered in any order, but the **match request method get** command is matched first.

```plaintext
match request header host length gt 100
    reset
match request method get
    log
```

If an action drops a packet, then no further actions are performed in the inspection policy map. For example, if the first action is to reset the connection, then it will never match any further **match** or **class** commands. If the first action is to log the packet, then a second action, such as resetting the connection, can occur. (You can configure both the **reset** (or **drop-connection**, and so on.) and the **log** action for the same **match** or **class** command, in which case the packet is logged before it is reset for a given match.)

If a packet matches multiple **match** or **class** commands that are the same, then they are matched in the order they appear in the policy map. For example, for a packet with the header length of 1001, it will match the first command below, and be logged, and then will match the second command and be reset. If you reverse the order of the two **match** commands, then the packet will be dropped and the connection reset before it can match the second **match** command; it will never be logged.

```plaintext
match request header length gt 100
    log
match request header length gt 1000
    reset
```

A class map is determined to be the same type as another class map or **match** command based on the lowest priority **match** command in the class map (the priority is based on the internal rules). If a class map has the same type of lowest priority **match** command as another class map, then the class maps are matched according to the order they are added to the policy map. If the lowest priority command for each class map is different, then the class map with the higher priority **match** command is matched first. For example, the following three class maps contain two types of **match** commands: **match request-cmd** (higher priority) and **match filename** (lower priority). The ftp3 class map includes both commands, but it is ranked according to the lowest priority command, **match filename**. The ftp1 class map includes the highest priority command, so it is matched first, regardless of the order in the policy map. The ftp3 class map is ranked as being of the same priority as the ftp2 class map, which also contains the **match filename** command. They are matched according to the order in the policy map: ftp3 and then ftp2.

```plaintext
class-map type inspect ftp match-all ftp1
    match request-cmd get
class-map type inspect ftp match-all ftp2
    match filename regex abc
class-map type inspect ftp match-all ftp3
    match request-cmd get
    match filename regex abc

policy-map type inspect ftp ftp
    class ftp3
        log
    class ftp2
        log
    class ftp1
        log
```
## Defining Actions in an Inspection Policy Map

### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** | (Optional) Create an inspection class map.  
See the “Identifying Traffic in an Inspection Class Map” section on page 31-6. Alternatively, you can identify the traffic directly within the policy map. |
| **Step 2** |  
**policy-map type inspect application**  
**policy_map_name**  
**Example:**  
hostname(config)# policy-map type inspect  
http http_policy  
Creates the inspection policy map. See the “Configuring Application Layer Protocol Inspection” section on page 39-6 for a list of applications that support inspection policy maps.  
The *policy_map_name* argument is the name of the policy map up to 40 characters in length. All types of policy maps use the same name space, so you cannot reuse a name already used by another type of policy map. The CLI enters policy-map configuration mode. |
| **Step 3** | Specify the traffic on which you want to perform actions using one of the following methods:  
**class class_map_name**  
**Example:**  
hostname(config-pmap)# class http_traffic  
hostname(config-pmap-c)#  
Specifies the inspection class map that you created in the “Identifying Traffic in an Inspection Class Map” section on page 31-6.  
Not all applications support inspection class maps. |
| | Specify traffic directly in the policy map using one of the **match** commands described for each application in the inspection chapter.  
**Example:**  
hostname(config-pmap)# match req-resp  
content-type mismatch  
hostname(config-pmap-c)#  
If you use a **match not** command, then any traffic that matches the criterion in the **match not** command does not have the action applied. |
Chapter 31  Configuring Special Actions for Application Inspections (Inspection Policy Map)

Defining Actions in an Inspection Policy Map

**Step 4**

### Command

```
[[drop [send-protocol-error] |
drop-connection [send-protocol-error] |
mask | reset] [log] | rate-limit
message_rate]
```

### Purpose

Specifies the action you want to perform on the matching traffic. Not all options are available for each application. Other actions specific to the application might also be available. See the appropriate inspection chapter for the exact options available.

- **drop**—Drops all packets that match.
- **send-protocol-error**—Sends a protocol error message.
- **drop-connection**—Drops the packet and closes the connection.
- **mask**—Masks out the matching portion of the packet.
- **reset**—Drops the packet, closes the connection, and sends a TCP reset to the server and/or client.
- **log**—Sends a system log message. You can use log alone or with one of the other keywords.
- **rate-limit**—Limits the rate of messages.

### Example:

```
hostname(config-pmap-c)# drop-connection
log
```

**Step 5**

### parameters

### Example:

```
hostname(config-pmap)# parameters
hostname(config-pmap-p)#
```

Configures parameters that affect the inspection engine. The CLI enters parameters configuration mode. For the parameters available for each application, see the appropriate inspection chapter.

### Examples

The following is an example of an HTTP inspection policy map and the related class maps. This policy map is activated by the Layer 3/4 policy map, which is enabled by the service policy.

```
hostname(config)# regex url_example example\.com
hostname(config)# regex url_example2 example2\.com
hostname(config)# class-map type regex match-any URLs
hostname(config-cmap)# match regex url_example
hostname(config-cmap)# match regex url_example2

hostname(config-cmap)# class-map type inspect http match-all http-traffic
hostname(config-cmap)# match req-resp content-type mismatch
hostname(config-cmap)# match request body length gt 1000
hostname(config-cmap)# match not request uri regex class URLs

hostname(config-cmap)# policy-map type inspect http http-map1
hostname(config-pmap)# class http-traffic
hostname(config-pmap-c)# drop-connection log
hostname(config-pmap-c)# match req-resp content-type mismatch
hostname(config-pmap-c)# reset log
hostname(config-pmap-c)# parameters
hostname(config-pmap-p)## protocol-violation action log

hostname(config-pmap-p)# policy-map test
hostname(config-pmap)# class test (a Layer 3/4 class map not shown)
hostname(config-pmap-c)# inspect http http-map1
hostname(config-pmap-c)# service-policy test interface outside
```
Identifying Traffic in an Inspection Class Map

This type of class map allows you to match criteria that is specific to an application. For example, for DNS traffic, you can match the domain name in a DNS query.

A class map groups multiple traffic matches (in a match-all class map), or lets you match any of a list of matches (in a match-any class map). The difference between creating a class map and defining the traffic match directly in the inspection policy map is that the class map lets you group multiple match commands, and you can reuse class maps. For the traffic that you identify in this class map, you can specify actions such as dropping, resetting, and/or logging the connection in the inspection policy map. If you want to perform different actions on different types of traffic, you should identify the traffic directly in the policy map.

Restrictions

Not all applications support inspection class maps. See the CLI help for **class-map type inspect** for a list of supported applications.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>(Optional) Create a regular expression. See the “Creating a Regular Expression” section on page 12-12 and the “Creating a Regular Expression Class Map” section on page 12-15.</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>**class-map type inspect application [match-all</td>
</tr>
</tbody>
</table>

Example:

hostname(config)# class-map type inspect http http_traffic
hostname(config-cmap)#
### Step 3
(Optional)

**description string**

**Example:**
```
hostname(config-cmap)# description All UDP traffic
```

### Step 4
Define the traffic to include in the class by entering one or more **match** commands available for your application.

To specify traffic that should not match the class map, use the **match not** command. For example, if the **match not** command specifies the string “example.com,” then any traffic that includes “example.com” does not match the class map.

To see the **match** commands available for each application, see the appropriate inspection chapter.

### Examples

The following example creates an HTTP class map that must match all criteria:

```
hostname(config-cmap)# class-map type inspect http match-all http-traffic
hostname(config-cmap)# match req-resp content-type mismatch
hostname(config-cmap)# match request body length gt 1000
hostname(config-cmap)# match not request uri regex class URLs
```

The following example creates an HTTP class map that can match any of the criteria:

```
hostname(config-cmap)# class-map type inspect http match-any monitor-http
hostname(config-cmap)# match request method get
hostname(config-cmap)# match request method put
hostname(config-cmap)# match request method post
```

### Where to Go Next

To use an inspection policy, see Chapter 30, “Configuring a Service Policy Using the Modular Policy Framework.”
PART 9

Configuring Access Control
CHAPTER 32

Configuring Access Rules

This chapter describes how to control network access through the ASASM using access rules and includes the following sections:

- Information About Access Rules, page 32-1
- Licensing Requirements for Access Rules, page 32-6
- Prerequisites, page 32-6
- Guidelines and Limitations, page 32-7
- Default Settings, page 32-7
- Configuring Access Rules, page 32-7
- Monitoring Access Rules, page 32-8
- Configuration Examples for Permitting or Denying Network Access, page 32-9
- Feature History for Access Rules, page 32-10

Note

You use access rules to control network access in both routed and transparent firewall modes. In transparent mode, you can use both access rules (for Layer 3 traffic) and EtherType rules (for Layer 2 traffic).

To access the ASASM interface for management access, you do not also need an access rule allowing the host IP address. You only need to configure management access according to Chapter 34, “Configuring Management Access.”

Information About Access Rules

You create an access rule by applying an extended or EtherType access list to an interface or globally for all interfaces. You can use access rules in routed and transparent firewall mode to control IP traffic. An access rule permits or denies traffic based on the protocol, a source and destination IP address or network, and optionally the source and destination ports.

For transparent mode only, an EtherType rule controls network access for non-IP traffic. An EtherType rule permits or denies traffic based on the EtherType.

This section includes the following topics:

- General Information About Rules, page 32-2
- Information About Extended Access Rules, page 32-4
General Information About Rules

This section describes information for both access rules and EtherType rules, and it includes the following topics:

- Implicit Permits, page 32-2
- Using Access Rules and EtherType Rules on the Same Interface, page 32-2
- Implicit Deny, page 32-3
- Inbound and Outbound Rules, page 32-3

Implicit Permits

For routed mode, the following types of traffic are allowed through by default:

- IPv4 traffic from a higher security interface to a lower security interface.
- IPv6 traffic from a higher security interface to a lower security interface.

For transparent mode, the following types of traffic are allowed through by default:

- IPv4 traffic from a higher security interface to a lower security interface.
- IPv6 traffic from a higher security interface to a lower security interface.
- ARPs in both directions.

Note: ARP traffic can be controlled by ARP inspection, but cannot be controlled by an access rule.

- BPDUs in both directions.

For other traffic, you need to use either an extended access rule (IPv4), an IPv6 access rule (IPv6), or an EtherType rule (non-IPv4/IPv6).

Information About Interface Access Rules and Global Access Rules

You can apply an access rule to a specific interface, or you can apply an access rule globally to all interfaces. You can configure global access rules in conjunction with interface access rules, in which case, the specific interface access rules are always processed before the general global access rules.

Note: Global access rules apply only to inbound traffic. See the “Inbound and Outbound Rules” section on page 32-3.

Using Access Rules and EtherType Rules on the Same Interface

You can apply one access rule and one EtherType rule to each direction of an interface.
**Implicit Deny**

Access lists have an implicit deny at the end of the list, so unless you explicitly permit it, traffic cannot pass. For example, if you want to allow all users to access a network through the ASASM except for particular addresses, then you need to deny the particular addresses and then permit all others.

For EtherType access lists, the implicit deny at the end of the access list does not affect IP traffic or ARPs; for example, if you allow EtherType 8037, the implicit deny at the end of the access list does not now block any IP traffic that you previously allowed with an extended access list (or implicitly allowed from a high security interface to a low security interface). However, if you explicitly deny all traffic with an EtherType ACE, then IP and ARP traffic is denied.

If you configure a global access rule, then the implicit deny comes after the global rule is processed. See the following order of operations:

1. Interface access rule.
2. Global access rule.
3. Implicit deny.

**Inbound and Outbound Rules**

The ASASM supports two types of access rules:

- **Inbound**—Inbound access rules apply to traffic as it enters an interface. Global access rules are always inbound.
- **Outbound**—Outbound access rules apply to traffic as it exits an interface.

---

**Note**

“Inbound” and “outbound” refer to the application of an access list on an interface, either to traffic entering the ASASM on an interface or traffic exiting the ASASM on an interface. These terms do not refer to the movement of traffic from a lower security interface to a higher security interface, commonly known as inbound, or from a higher to lower interface, commonly known as outbound.

An outbound access list is useful, for example, if you want to allow only certain hosts on the inside networks to access a web server on the outside network. Rather than creating multiple inbound access lists to restrict access, you can create a single outbound access list that allows only the specified hosts. (See Figure 32-1.) The outbound access list prevents any other hosts from reaching the outside network.
Information About Access Rules

This section describes information about extended access rules and includes the following topics:

- Access Rules for Returning Traffic, page 32-4
- Allowing Broadcast and Multicast Traffic through the Transparent Firewall Using Access Rules, page 32-5
- Management Access Rules, page 32-5

Access Rules for Returning Traffic

For TCP and UDP connections for both routed and transparent mode, you do not need an access rule to allow returning traffic because the ASASM allows all returning traffic for established, bidirectional connections.
For connectionless protocols such as ICMP, however, the ASASM establishes unidirectional sessions, so you either need access rules to allow ICMP in both directions (by applying access lists to the source and destination interfaces), or you need to enable the ICMP inspection engine. The ICMP inspection engine treats ICMP sessions as bidirectional connections. To control ping, specify `echo-reply` (0) (ASASM to host) or `echo` (8) (host to ASASM).

## Allowing Broadcast and Multicast Traffic through the Transparent Firewall Using Access Rules

In routed firewall mode, broadcast and multicast traffic is blocked even if you allow it in an access rule, including unsupported dynamic routing protocols and DHCP (unless you configure DHCP relay). Transparent firewall mode can allow any IP traffic through. This feature is especially useful in multiple context mode, which does not allow dynamic routing, for example.

---

**Note**

Because these special types of traffic are connectionless, you need to apply an extended access list to both interfaces, so returning traffic is allowed through.

Table 32-1 lists common traffic types that you can allow through the transparent firewall.

### Table 32-1  Transparent Firewall Special Traffic

<table>
<thead>
<tr>
<th>Traffic Type</th>
<th>Protocol or Port</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>DHCP</td>
<td>UDP ports 67 and 68</td>
<td>If you enable the DHCP server, then the ASASM does not pass DHCP packets.</td>
</tr>
<tr>
<td>EIGRP</td>
<td>Protocol 88</td>
<td>—</td>
</tr>
<tr>
<td>OSPF</td>
<td>Protocol 89</td>
<td>—</td>
</tr>
<tr>
<td>Multicast streams</td>
<td>The UDP ports vary depending on the application.</td>
<td>Multicast streams are always destined to a Class D address (224.0.0.0 to 239.x.x.x).</td>
</tr>
<tr>
<td>RIP (v1 or v2)</td>
<td>UDP port 520</td>
<td>—</td>
</tr>
</tbody>
</table>

## Management Access Rules

You can configure access rules that control management traffic destined to the ASASM. Access control rules for to-the-box management traffic (defined by such commands as `http`, `ssh`, or `telnet`) have higher precedence than an management access rule applied with the `control-plane` option. Therefore, such permitted management traffic will be allowed to come in even if explicitly denied by the to-the-box access list.

## Information About EtherType Rules

This section describes EtherType rules and includes the following topics:

- Supported EtherTypes and Other Traffic, page 32-6
- Access Rules for Returning Traffic, page 32-6
- Allowing MPLS, page 32-6
Supported EtherTypes and Other Traffic

An EtherType rule controls the following:

- EtherType identified by a 16-bit hexadecimal number, including common types IPX and MPLS unicast or multicast.
- Ethernet V2 frames.
- BPDUs, which are permitted by default. BPDUs are SNAP-encapsulated, and the ASASM is designed to specifically handle BPDUs.
- Trunk port (Cisco proprietary) BPDUs. Trunk BPDUs have VLAN information inside the payload, so the ASASM modifies the payload with the outgoing VLAN if you allow BPDUs.

The following types of traffic are not supported:

- 802.3-formatted frames—These frames are not handled by the rule because they use a length field as opposed to a type field.

Access Rules for Returning Traffic

Because EtherTypes are connectionless, you need to apply the rule to both interfaces if you want traffic to pass in both directions.

Allowing MPLS

If you allow MPLS, ensure that Label Distribution Protocol and Tag Distribution Protocol TCP connections are established through the ASASM by configuring both MPLS routers connected to the ASASM to use the IP address on the ASASM interface as the router-id for LDP or TDP sessions. (LDP and TDP allow MPLS routers to negotiate the labels (addresses) used to forward packets.)

On Cisco IOS routers, enter the appropriate command for your protocol, LDP or TDP. The interface is the interface connected to the ASASM.

```
hostname(config)# mpls ldp router-id interface force
```

Or

```
hostname(config)# tag-switching tdp router-id interface force
```

Licensing Requirements for Access Rules

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

Prerequisites

Before you can create an access rule, create the access list. See Chapter 14, “Adding an Extended Access List,” and Chapter 15, “Adding an EtherType Access List,” for more information.
Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

**Context Mode Guidelines**
Supported in single and multiple context mode.

**Firewall Mode Guidelines**
Supported in routed and transparent firewall modes.

**IPv6 Guidelines**
Supports IPv6.

**Per-User Access List Guidelines**
- If there is no per-user access list associated with a packet, the interface access rule is applied.
- The per-user access list uses the value in the `timeout uauth` command, but it can be overridden by the AAA per-user session timeout value.
- If traffic is denied because of a per-user access list, syslog message 109025 is logged. If traffic is permitted, no syslog message is generated. The `log` option in the per-user access list has no effect.

**Default Settings**

See the “Implicit Permits” section on page 32-2.

**Configuring Access Rules**

To apply an access rule, perform the following steps.
Monitoring Access Rules

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| access-group access_list \{(in | out) interface interface_name \[per-user-override | control-plane] | global\} | Binds an access list to an interface or applies it globally. Specify the extended, EtherType, or IPv6 access list name. You can configure one access-group command per access list type per interface. You cannot reference empty access lists or access lists that contain only a remark. For an interface-specific rule:

- The in keyword applies the access list to inbound traffic. The out keyword applies the access list to the outbound traffic.
- Specify the interface name.
- The per-user-override keyword (for inbound access lists only) allows dynamic user access lists that are downloaded for user authorization to override the access list assigned to the interface. For example, if the interface access list denies all traffic from 10.0.0.0, but the dynamic access list permits all traffic from 10.0.0.0, then the dynamic access list overrides the interface access list for that user. See the “Configuring RADIUS Authorization” section on page 35-14 for more information about per-user access lists. See also the “Per-User Access List Guidelines” section on page 32-7.
- The control-plane keyword specifies if the rule is for to-the-box traffic.

For a global rule, specify the global keyword to apply the access list to the inbound direction of all interfaces.

Example:
hostname(config)# access-group acl_out in interface outside

Examples

The following example shows how to use the access-group command:

hostname(config)# access-list acl_out permit tcp any host 209.165.201.3 eq 80
hostname(config)# access-group acl_out in interface outside

The access-list command lets any host access the global address using port 80. The access-group command specifies that the access-list command applies to traffic entering the outside interface.

Monitoring Access Rules

To monitor network access, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show running-config access-group</td>
<td>Displays the current access list bound to the interfaces.</td>
</tr>
</tbody>
</table>
Configuration Examples for Permitting or Denying Network Access

This section includes typical configuration examples for permitting or denying network access.

The following example illustrates the commands required to enable access to an inside web server with the IP address 209.165.201.12. (This IP address is the real address, not the visible on the outside interface after NAT.)

```
hostname(config)# access-list ACL_OUT extended permit tcp any host 209.165.201.12 eq www
hostname(config)# access-group ACL_OUT in interface outside
```

The following example allows all hosts to communicate between the inside and hr networks but only specific hosts to access the outside network:

```
hostname(config)# access-list ANY extended permit ip any any
hostname(config)# access-list OUT extended permit ip host 209.168.200.3 any
hostname(config)# access-list OUT extended permit ip host 209.168.200.4 any
hostname(config)# access-group ANY in interface inside
hostname(config)# access-group ANY in interface hr
hostname(config)# access-group OUT out interface outside
```

For example, the following sample access list allows common EtherTypes originating on the inside interface:

```
hostname(config)# access-list ETHER ethertype permit ipx
hostname(config)# access-list ETHER ethertype permit mpls-unicast
hostname(config)# access-group ETHER in interface inside
```

The following example allows some EtherTypes through the ASASM, but it denies all others:

```
hostname(config)# access-list ETHER ethertype permit 0x1234
hostname(config)# access-list ETHER ethertype permit mpls-unicast
hostname(config)# access-group ETHER in interface inside
```

The following example denies traffic with EtherType 0x1256 but allows all others on both interfaces:

```
hostname(config)# access-list nonIP ethertype deny 1256
hostname(config)# access-list nonIP ethertype permit any
hostname(config)# access-group ETHER in interface inside
```

The following example uses object groups to permit specific traffic on the inside interface:

```
!
hostname(config)# object-group service myaclog
hostname(config-service)# service-object tcp source range 2000 3000
hostname(config-service)# service-object tcp source range 3000 3010 destination
hostname(config-service)# service-object ipsec
hostname(config-service)# service-object udp destination range 1002 1006
hostname(config-service)# service-object icmp echo
```

```
hostname(config)# access-list outsideacl extended permit object-group myaclog inside any
```
Feature History for Access Rules

Table 32-2 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interface access rules</td>
<td>7.0(1)</td>
<td>Controlling network access through the ASASM using access lists. We introduced the following command: <code>access-group</code>.</td>
</tr>
<tr>
<td>Global access rules</td>
<td>8.3(1)</td>
<td>Global access rules were introduced. We modified the following command: <code>access-group</code>.</td>
</tr>
</tbody>
</table>
CHAPTER 33

Configuring AAA Servers and the Local Database

This chapter describes support for authentication, authorization, and accounting (AAA, pronounced “triple A”), and how to configure AAA servers and the local database.

The chapter includes the following sections:

- Information About AAA, page 33-1
- Licensing Requirements for AAA Servers, page 33-9
- Guidelines and Limitations, page 33-9
- Configuring AAA, page 33-10
- Monitoring AAA Servers, page 33-27
- Additional References, page 33-28
- Feature History for AAA Servers, page 33-28

Information About AAA

AAA enables the ASASM to determine who the user is (authentication), what the user can do (authorization), and what the user did (accounting).

AAA provides an extra level of protection and control for user access than using access lists alone. For example, you can create an access list allowing all outside users to access Telnet on a server on the DMZ network. If you want only some users to access the server and you might not always know IP addresses of these users, you can enable AAA to allow only authenticated and/or authorized users to connect through the ASASM. (The Telnet server enforces authentication, too; the ASASM prevents unauthorized users from attempting to access the server.)

You can use authentication alone or with authorization and accounting. Authorization always requires a user to be authenticated first. You can use accounting alone, or with authentication and authorization.

This section includes the following topics:

- Information About Authentication, page 33-2
- Information About Authorization, page 33-2
- Information About Accounting, page 33-3
- Summary of Server Support, page 33-3
- RADIUS Server Support, page 33-3
- TACACS+ Server Support, page 33-5
Information About AAA

Authentication controls access by requiring valid user credentials, which are usually a username and password. You can configure the ASASM to authenticate the following items:

- All administrative connections to the ASASM, including the following sessions:
  - Telnet
  - SSH
  - Serial console
  - ASDM using HTTPS
- The `enable` command
- Network access

Information About Authorization

Authorization controls access per user after users are authenticated. You can configure the ASASM to authorize the following items:

- Management commands
- Network access

Authorization controls the services and commands that are available to each authenticated user. If you did not enable authorization, authentication alone would provide the same access to services for all authenticated users.

If you need the control that authorization provides, you can configure a broad authentication rule, and then have a detailed authorization configuration. For example, you can authenticate inside users who try to access any server on the outside network and then limit the outside servers that a particular user can access using authorization.

The ASASM caches the first 16 authorization requests per user, so if the user accesses the same services during the current authentication session, the ASASM does not resend the request to the authorization server.
Information About Accounting

Accounting tracks traffic that passes through the ASASM, enabling you to have a record of user activity. If you enable authentication for that traffic, you can account for traffic per user. If you do not authenticate the traffic, you can account for traffic per IP address. Accounting information includes session start and stop times, username, the number of bytes that pass through the ASASM for the session, the service used, and the duration of each session.

Summary of Server Support

Table 33-1 summarizes the support for each AAA service by each AAA server type, including the local database. For more information about support for a specific AAA server type, see the topics following the table.

Table 33-1 Summary of AAA Support

<table>
<thead>
<tr>
<th>AAA Service</th>
<th>Database Type</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Local</td>
<td>RADIUS</td>
<td>TACACS+</td>
<td>SDI (RSA)</td>
<td>NT</td>
<td>Kerberos</td>
<td>LDAP</td>
</tr>
<tr>
<td>Authentication of...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Firewall sessions</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Administrators</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Authorization of...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Firewall sessions</td>
<td>No</td>
<td>Yes²</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Administrators</td>
<td>Yes³</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Accounting of...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Firewall sessions</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Administrators</td>
<td>No</td>
<td>Yes⁴</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>

1. RSA/SDI is supported for ASDM HTTP administrative access with ASA 5500 software version 8.2(1) or later.
2. For firewall sessions, RADIUS authorization is supported with user-specific access lists only, which are received or specified in a RADIUS authentication response.
3. Local command authorization is supported by privilege level only.
4. Command accounting is available for TACACS+ only.

In addition to the native protocol authentication listed in Table 33-1, the ASASM supports proxying authentication. For example, the ASASM can proxy to an RSA/SDI and/or LDAP server via a RADIUS server. Authentication via digital certificates and/or digital certificates with the AAA combinations listed in the table are also supported.

RADIUS Server Support

The ASASM supports the following RFC-compliant RADIUS servers for AAA:

- Cisco Secure ACS 3.2, 4.0, 4.1, 4.2, and 5.x
- Cisco Identity Services Engine (ISE)
Authentication Methods

The ASASM supports the following authentication methods with RADIUS:

- PAP—For all connection types.
- Authentication Proxy modes—including RADIUS to Active Directory, RADIUS to RSA/SDI, RADIUS to Token-server, and RSA/SDI to RADIUS connections.

Attribute Support

The ASASM supports the following sets of RADIUS attributes:

- Authentication attributes defined in RFC 2138.
- Accounting attributes defined in RFC 2139.
- RADIUS attributes for tunneled protocol support, defined in RFC 2868.
- Cisco IOS Vendor-Specific Attributes (VSAs), identified by RADIUS vendor ID 9.
- Microsoft VSAs, defined in RFC 2548.
- Cisco VSA (Cisco-Priv-Level), which provides a standard 0-15 numeric ranking of privileges, with 1 being the lowest level and 15 being the highest level. A zero level indicates no privileges. The first level (login) allows privileged EXEC access for the commands available at this level. The second level (enable) allows CLI configuration privileges.
- A list of attributes is available at the following URL: http://www.cisco.com/en/US/docs/security/asa/asa84/configuration/guide/ref_extserver.html#wp1605508

RADIUS Authorization Functions

The ASASM can use RADIUS servers for user authorization of VPN remote access and firewall cut-through-proxy sessions using dynamic access lists or access list names per user. To implement dynamic access lists, you must configure the RADIUS server to support it. When the user authenticates,
the RADIUS server sends a downloadable access list or access list name to the ASASM. Access to a given service is either permitted or denied by the access list. The ASASM deletes the access list when the authentication session expires.

In addition to access lists, the ASASM supports many other attributes for authorization and setting of permissions for VPN remote access and firewall cut-through proxy sessions. For a complete list of authorization attributes, see the following URL:

TACACS+ Server Support

The ASASM supports TACACS+ authentication with ASCII, PAP, CHAP, and MS-CHAPv1.

RSA/SDI Server Support

The RSA SecureID servers are also known as SDI servers.

This section includes the following topics:

- RSA/SDI Version Support, page 33-5
- RSA/SDI Primary and Replica Servers, page 33-6

RSA/SDI Version Support

The ASASM supports SDI Versions 5.x, 6.x, and 7.x. SDI uses the concepts of an SDI primary and SDI replica servers. Each primary and its replicas share a single node secret file. The node secret file has its name based on the hexadecimal value of the ACE or Server IP address, with .sdi appended.

A version 5.x, 6.x, or 7.x SDI server that you configure on the ASASM can be either the primary or any one of the replicas. See the “RSA/SDI Primary and Replica Servers” section on page 33-6 for information about how the SDI agent selects servers to authenticate users.

Two-step Authentication Process

SDI Versions 5.x, 6.x, or 7.x use a two-step process to prevent an intruder from capturing information from an RSA SecurID authentication request and using it to authenticate to another server. The agent first sends a lock request to the SecurID server before sending the user authentication request. The server
locks the username, preventing another (replica) server from accepting it. This action means that the same user cannot authenticate to two ASASMs using the same authentication servers simultaneously. After a successful username lock, the ASASM sends the passcode.

**RSA/SDI Primary and Replica Servers**

The ASASM obtains the server list when the first user authenticates to the configured server, which can be either a primary or a replica. The ASASM then assigns priorities to each of the servers on the list, and subsequent server selection is derived at random from those assigned priorities. The highest priority servers have a higher likelihood of being selected.

**NT Server Support**

The ASASM supports Microsoft Windows server operating systems that support NTLM Version 1, collectively referred to as NT servers.

*Note*
NT servers have a maximum length of 14 characters for user passwords. Longer passwords are truncated, which is a limitation of NTLM Version 1.

**Kerberos Server Support**

The ASASM supports 3DES, DES, and RC4 encryption types.

For a simple Kerberos server configuration example, see Example 33-2 on page 33-16.

**LDAP Server Support**

The ASASM supports LDAP. This section includes the following topics:

- Authentication with LDAP, page 33-6
- LDAP Server Types, page 33-7

**Authentication with LDAP**

During authentication, the ASASM acts as a client proxy to the LDAP server for the user, and authenticates to the LDAP server in either plain text or by using the SASL protocol. By default, the ASASM passes authentication parameters, usually a username and password, to the LDAP server in plain text.

The ASASM supports the following SASL mechanisms, listed in order of increasing strength:

- Digest-MD5—The ASASM responds to the LDAP server with an MD5 value computed from the username and password.
- Kerberos—The ASASM responds to the LDAP server by sending the username and realm using the GSSAPI Kerberos mechanism.
You can configure the ASASM and LDAP server to support any combination of these SASL mechanisms. If you configure multiple mechanisms, the ASASM retrieves the list of SASL mechanisms that are configured on the server and sets the authentication mechanism to the strongest mechanism configured on both the ASASM and the server. For example, if both the LDAP server and the ASASM support both mechanisms, the ASASM selects Kerberos, the stronger of the mechanisms.

When user LDAP authentication has succeeded, the LDAP server returns the attributes for the authenticated user. For VPN authentication, these attributes generally include authorization data that is applied to the VPN session. Thus, using LDAP accomplishes authentication and authorization in a single step.

**LDAP Server Types**

The ASASM supports LDAP version 3 and is compatible with the Sun Microsystems JAVA System Directory Server (formerly named the Sun ONE Directory Server), the Microsoft Active Directory, Novell, OpenLDAP, and other LDAPv3 directory servers.

By default, the ASASM auto-detects whether it is connected to Microsoft Active Directory, Sun LDAP, Novell, OpenLDAP, or a generic LDAPv3 directory server. However, if auto-detection fails to determine the LDAP server type, and you know the server is either a Microsoft, Sun or generic LDAP server, you can manually configure the server type.

When configuring the server type, note the following guidelines:

- The DN configured on the ASASM to access a Sun directory server must be able to access the default password policy on that server. We recommend using the directory administrator, or a user with directory administrator privileges, as the DN. Alternatively, you can place an ACL on the default password policy.
- You must configure LDAP over SSL to enable password management with Microsoft Active Directory and Sun servers.
- The ASASM does not support password management with Novell, OpenLDAP, and other LDAPv3 directory servers.
- The ASASM uses the Login Distinguished Name (DN) and Login Password to establish a trust relationship (bind) with an LDAP server. For more information, see the “Binding the ASASM to the LDAP Server” section on page C-4.

**Local Database Support, Including as a Falback Method**

The ASASM maintains a local database that you can populate with user profiles.

The local database can act as a fallback method for several functions. This behavior is designed to help you prevent accidental lockout from the ASASM.

For users who need fallback support, we recommend that their usernames and passwords in the local database match their usernames and passwords on the AAA servers. This practice provides transparent fallback support. Because the user cannot determine whether a AAA server or the local database is providing the service, using usernames and passwords on AAA servers that are different than the usernames and passwords in the local database means that the user cannot be certain which username and password should be given.

The local database supports the following fallback functions:

- Console and enable password authentication—If the servers in the group are all unavailable, the ASASM uses the local database to authenticate administrative access, which can also include enable password authentication.
• Command authorization—If the TACACS+ servers in the group are all unavailable, the local database is used to authorize commands based on privilege levels.

How Fallback Works with Multiple Servers in a Group

If you configure multiple servers in a server group and you enable fallback to the local database for the server group, fallback occurs when no server in the group responds to the authentication request from the ASASM. To illustrate, consider this scenario:

You configure an LDAP server group with two Active Directory servers, server 1 and server 2, in that order. When the remote user logs in, the ASASM attempts to authenticate to server 1.

If server 1 responds with an authentication failure (such as user not found), the ASASM does not attempt to authenticate to server 2.

If server 1 does not respond within the timeout period (or the number of authentication attempts exceeds the configured maximum), the ASASM tries server 2.

If both servers in the group do not respond, and the ASASM is configured to fall back to the local database, the ASASM tries to authenticate to the local database.

Using Certificates and User Login Credentials

The following section describes the different methods of using certificates and user login credentials (username and password) for authentication and authorization.

In all cases, LDAP authorization does not use the password as a credential. RADIUS authorization uses either a common password for all users or the username as a password.

This section includes the following topics:

• Using User Login Credentials, page 33-8
• Using Certificates, page 33-9

Using User Login Credentials

The default method for authentication and authorization uses the user login credentials.

• Authentication
  – Enabled by the authentication server group setting in the tunnel group (also called ASDM Connection Profile)
  – Uses the username and password as credentials

• Authorization
  – Enabled by the authorization server group setting in the tunnel group (also called ASDM Connection Profile)
  – Uses the username as a credential
Using Certificates

If user digital certificates are configured, the ASASM first validates the certificate. It does not, however, use any of the DN fields from certificates as a username for the authentication.

If both authentication and authorization are enabled, the ASASM uses the user login credentials for both user authentication and authorization.

- **Authentication**
  - Enabled by the authentication server group setting
  - Uses the username and password as credentials
- **Authorization**
  - Enabled by the authorization server group setting
  - Uses the username as a credential

If authentication is disabled and authorization is enabled, the ASASM uses the primary DN field for authorization.

- **Authentication**
  - DISABLED (set to None) by the authentication server group setting
  - No credentials used
- **Authorization**
  - Enabled by the authorization server group setting
  - Uses the username value of the certificate primary DN field as a credential

If the primary DN field is not present in the certificate, the ASASM uses the secondary DN field value as the username for the authorization request.

For example, consider a user certificate that includes the following Subject DN fields and values:

\[Cn=anyuser,OU=sales;O=XYZCorporation;L=boston;S=mass;C=us;ea=anyuser@example.com\]

If the Primary DN = EA (E-mail Address) and the Secondary DN = CN (Common Name), then the username used in the authorization request would be anyuser@example.com.

Licensing Requirements for AAA Servers

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

Guidelines and Limitations

This section includes the guidelines and limitations for this feature.
Context Mode Guidelines
Supported in single and multiple context mode.

Firewall Mode Guidelines
Supported in routed and transparent firewall mode.

IPv6 Guidelines
Supports IPv6.

Additional Guidelines
The *username* command has two versions: one for 8.4(3) and earlier and one for 8.4(4.1) and later. See the command reference for more information.

Configuring AAA

This section includes the following topics:
- Configuring AAA Server Groups, page 33-11
- Configuring LDAP Attribute Maps, page 33-16
- Adding a User Account to the Local Database, page 33-19
- Managing User Passwords, page 33-23
- Changing User Passwords, page 33-24
- Authenticating Users with a Public Key for SSH, page 33-25
- Differentiating User Roles Using AAA, page 33-25

Task Flow for Configuring AAA

<table>
<thead>
<tr>
<th>Step 1</th>
<th>Do one or both of the following:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• Add a AAA server group. See the “Configuring AAA Server Groups” section on page 33-11.</td>
</tr>
<tr>
<td></td>
<td>• Add a user to the local database. See the “Adding a User Account to the Local Database” section on page 33-19.</td>
</tr>
</tbody>
</table>

| Step 2 | For an LDAP server, configure LDAP attribute maps. See the “Configuring LDAP Attribute Maps” section on page 33-16. |

| Step 3 | For an administrator, specify the password policy attributes for users. See the “Managing User Passwords” section on page 33-23. |

| Step 4 | (Optional) Users can change their own passwords. See the “Changing User Passwords” section on page 33-24. |

| Step 5 | (Optional) Users can authenticate with a public key. See the “Authenticating Users with a Public Key for SSH” section on page 33-25. |

| Step 6 | (Optional) Distinguish between administrative and remote-access users when they authenticate. See the “Differentiating User Roles Using AAA” section on page 33-25. |
Configuring AAA Server Groups

If you want to use an external AAA server for authentication, authorization, or accounting, you must first create at least one AAA server group per AAA protocol and add one or more servers to each group. You identify AAA server groups by name. Each server group is specific to one type of server: Kerberos, LDAP, NT, RADIUS, SDI, or TACACS+.

Guidelines

- You can have up to 100 server groups in single mode or 4 server groups per context in multiple mode.
- Each group can have up to 16 servers in single mode or 4 servers in multiple mode.
- When a user logs in, the servers are accessed one at a time, starting with the first server you specify in the configuration, until a server responds. If all servers in the group are unavailable, the ASASM tries the local database if you configured it as a fallback method (management authentication and authorization only). If you do not have a fallback method, the ASASM continues to try the AAA servers.
## Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> `aaa-server server_tag protocol {kerberos</td>
<td>ldap</td>
</tr>
</tbody>
</table>

Example:
- `hostname(config)# aaa-server servergroup1 protocol ldap`
- `hostname(config-aaa-server-group)#`
- `hostname(config)# aaa-server servergroup1 protocol radius interim-accounting-update`
- `hostname(config-aaa-server-group)# ad-agent-mode`

**Tip** Choose this option if users have trouble completing a VPN connection using clean access SSO, which might occur when making clientless or AnyConnect connections directly to the ASASM.

The `interim-accounting-update` option enables multi-session accounting for clientless SSL and AnyConnect sessions. If you choose this option, interim accounting records are sent to the RADIUS server in addition to the start and stop records.

The `ad-agent-mode` option specifies the shared secret between the ASASM and the AD agent, and indicates that a RADIUS server group includes AD agents that are not full-function RADIUS servers. Only a RADIUS server group that has been configured using the `ad-agent-mode` option can be associated with user identity. As a result, the `test aaa-server {authentication | authorization} aaa-server-group` command is not available when a RADIUS server group that is not configured using the `ad-agent-mode` option is specified.
### Chapter 33 Configuring AAA Servers and the Local Database

#### Configuring AAA

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>merge-dacl {before-avpair</td>
<td>after-avpair}</td>
</tr>
<tr>
<td>3</td>
<td>max-failed-attempts number</td>
<td>Specifies the maximum number of requests sent to a AAA server in the group before trying the next server. The <strong>number</strong> argument can range from 1 and 5. The default is 3.</td>
</tr>
</tbody>
</table>

#### Example:

**Step 2:**
```
hostname(config)# aaa-server servergroup1
protocol radius
hostname(config-aaa-server-group)# merge-dacl before-avpair
```

**Step 3:**
```
hostname(config-aaa-server-group)# max-failed-attempts 2
```
**Step 4**

```
reactivation-mode [depletion [deadtime minutes] | timed]
```

**Example:**
```
hostname(config-aaa-server-group)#
reactivation-mode deadtime 20
```

Specifies the method (reactivation policy) by which failed servers in a group are reactivated.

The **delegation** keyword reactivates failed servers only after all of the servers in the group are inactive.

The **deadtime minutes** keyword-argument pair specifies the amount of time in minutes, between 0 and 1440, that elapses between the disabling of the last server in the group and the subsequent reenabling of all servers. The default is 10 minutes.

The **timed** keyword reactivates failed servers after 30 seconds of down time.

**Step 5**

```
accounting-mode simultaneous
```

**Example:**
```
hostname(config-aaa-server-group)#
accounting-mode simultaneous
```

Sends accounting messages to all servers in the group (RADIUS or TACACS+ only).

To restore the default of sending messages only to the active server, enter the **accounting-mode single** command.

**Step 6**

```
aaa-server server_group [interface_name] host server_ip
```

**Example:**
```
hostname(config)# aaa-server servergroup1 outside host 10.10.1.1
```

Identifies the server and the AAA server group to which it belongs.

When you enter the **aaa-server host** command, you enter aaa-server host configuration mode. As needed, use host configuration mode commands to further configure the AAA server.

The commands in host configuration mode do not apply to all AAA server types. Table 33-2 lists the available commands, the server types to which they apply, and whether or not a new AAA server definition has a default value for that command. Where a command is applicable to the specified server type and no default value is provided (indicated by “—”), use the command to specify the value.

<table>
<thead>
<tr>
<th>Command</th>
<th>Applicable AAA Server Types</th>
<th>Default Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>accounting-port</td>
<td>RADIUS</td>
<td>1646</td>
<td></td>
</tr>
<tr>
<td>acl-netmask-convert</td>
<td>RADIUS</td>
<td>standard</td>
<td></td>
</tr>
<tr>
<td>authentication-port</td>
<td>RADIUS</td>
<td>1645</td>
<td></td>
</tr>
<tr>
<td>kerberosrealm</td>
<td>Kerberos</td>
<td>—</td>
<td></td>
</tr>
<tr>
<td>key</td>
<td>RADIUS</td>
<td>—</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TACACS+</td>
<td>—</td>
<td></td>
</tr>
<tr>
<td>ldap-attribute-map</td>
<td>LDAP</td>
<td>—</td>
<td></td>
</tr>
<tr>
<td>ldap-base-dn</td>
<td>LDAP</td>
<td>—</td>
<td></td>
</tr>
<tr>
<td>ldap-login-dn</td>
<td>LDAP</td>
<td>—</td>
<td></td>
</tr>
</tbody>
</table>

Table 33-2 **Host Mode Commands, Server Types, and Defaults**
Table 33-2  Host Mode Commands, Server Types, and Defaults (continued)

<table>
<thead>
<tr>
<th>Command</th>
<th>Applicable AAA Server Types</th>
<th>Default Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ldap-login-password</td>
<td>LDAP</td>
<td>—</td>
<td></td>
</tr>
<tr>
<td>ldap-naming-attribute</td>
<td>LDAP</td>
<td>—</td>
<td></td>
</tr>
<tr>
<td>ldap-over-ssl</td>
<td>LDAP</td>
<td>636</td>
<td>If not set, the ASASM uses sAMAccountName for LDAP requests. Whether using SASL or plain text, you can secure communications between the ASASM and the LDAP server with SSL. If you do not configure SASL, we strongly recommend that you secure LDAP communications with SSL.</td>
</tr>
<tr>
<td>ldap-scope</td>
<td>LDAP</td>
<td>—</td>
<td></td>
</tr>
<tr>
<td>mschapv2-capable</td>
<td>RADIUS</td>
<td>enabled</td>
<td></td>
</tr>
<tr>
<td>nt-auth-domain-controller</td>
<td>NT</td>
<td>—</td>
<td></td>
</tr>
<tr>
<td>radius-common-pw</td>
<td>RADIUS</td>
<td>—</td>
<td></td>
</tr>
<tr>
<td>retry-interval</td>
<td>Kerberos</td>
<td>10 seconds</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RADIUS</td>
<td>10 seconds</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SDI</td>
<td>10 seconds</td>
<td></td>
</tr>
<tr>
<td>sasl-mechanism</td>
<td>LDAP</td>
<td>—</td>
<td></td>
</tr>
<tr>
<td>server-port</td>
<td>Kerberos</td>
<td>88</td>
<td></td>
</tr>
<tr>
<td></td>
<td>LDAP</td>
<td>389</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NT</td>
<td>139</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SDI</td>
<td>5500</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TACACS+</td>
<td>49</td>
<td></td>
</tr>
<tr>
<td>server-type</td>
<td>LDAP</td>
<td>auto-discovery</td>
<td>If auto-detection fails to determine the LDAP server type, and you know the server is either a Microsoft, Sun or generic LDAP server, you can manually configure the server type.</td>
</tr>
<tr>
<td>timeout</td>
<td>All</td>
<td>10 seconds</td>
<td></td>
</tr>
</tbody>
</table>

Examples

Example 33-1 shows how to add one TACACS+ group with one primary and one backup server, one RADIUS group with a single server, and an NT domain server.

Example 33-1  Multiple AAA Server Groups and Servers

hostname(config)#  aaa-server AuthInbound protocol tacacs+
hostname(config-aaa-server-group)#  max-failed-attempts 2
hostname(config-aaa-server-group)#  reactivation-mode depletion deadtime 20
hostname(config-aaa-server-group)#  exit
hostname(config)#  aaa-server AuthInbound (inside) host 10.1.1.1
hostname(config-aaa-server-host)#  key TACPlusUauthKey
hostname(config-aaa-server-host)#  exit
hostname(config)#  aaa-server AuthInbound (inside) host 10.1.1.2
hostname(config-aaa-server-host)#  key TACPlusUauthKey2
hostname(config-aaa-server-host)#  exit
Example 33-2 shows how to configure a Kerberos AAA server group named watchdogs, add a AAA server to the group, and define the Kerberos realm for the server. Because Example 33-2 does not define a retry interval or the port that the Kerberos server listens to, the ASASM uses the default values for these two server-specific parameters. Table 33-2 lists the default values for all AAA server host mode commands.

Note

Kerberos realm names use numbers and upper-case letters only. Although the ASASM accepts lower-case letters for a realm name, it does not translate lower-case letters to upper-case letters. Be sure to use upper-case letters only.

Example 33-2  Kerberos Server Group and Server

hostname(config)# aaa-server watchdogs protocol kerberos
hostname(config-aaa-server-group)# aaa-server watchdogs host 192.168.3.4
hostname(config-aaa-server-host)# kerberos-realm EXAMPLE.COM
hostname(config-aaa-server-host)# exit
hostname(config)#

Configuring LDAP Attribute Maps

The ASASM can use an LDAP directory for authenticating VPN remote access users or firewall network access/cut-thru-proxy sessions and/or for setting policy permissions (also called authorization attributes), such as ACLs, bookmark lists, DNS or WINS settings, session timers, and so on. That is, you can set the key attributes that exist in a local group policy externally through an LDAP server.

The authorization process is accomplished by means of LDAP attribute maps (similar to a RADIUS dictionary that defines vendor-specific attributes), which translate the native LDAP user attributes to Cisco ASASM attribute names. You can then bind these attribute maps to LDAP servers or remove them, as needed. You can also show or clear attribute maps.

Guidelines

The ldap-attribute-map has a limitation with multi-valued attributes. For example, if a user is a memberOf of several AD groups and the ldap attribute map matches on more than one of them, the mapped value is chosen based on the alphabetization of the matched entries.

To use the attribute mapping features correctly, you need to understand Cisco LDAP attribute names and values, as well as the user-defined attribute names and values. For more information about LDAP attribute maps, see the “Active Directory/LDAP VPN Remote Access Authorization Examples” section on page C-16.
The names of frequently mapped Cisco LDAP attributes and the type of user-defined attributes that they would commonly be mapped to include the following:

- **IETF-Radius-Class (Group_Policy in ASASM version 8.2 and later)**—Sets the group policy based on the directory’s department or user group (for example, Microsoft Active Directory memberOf) attribute value. The group-policy attribute replaced the IETF-Radius-Class attribute with ASDM version 6.2/ASA version 8.2 or later.
- **IETF-Radius-Filter-Id**—An access control list or ACL applied to VPN clients, IPsec, and SSL.
- **IETF-Radius-Framed-IP-Address**—Assigns a static IP address assigned to a VPN remote access client, IPsec, and SSL.
- **Banner1**—Displays a text banner when the VPN remote access user logs in.
- **Tunneling-Protocols**—Allows or denies the VPN remote access session based on the access type.

Note

A single ldapattribute map may contain one or many attributes. You can only assign one ldap attribute to a specific LDAP server.

To map LDAP features correctly, perform the following steps:

**Detailed Steps**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 ldap attribute-map map-name</td>
<td>Creates an unpopulated LDAP attribute map table.</td>
</tr>
<tr>
<td>Example: hostname(config)# ldap attribute-map att_map_1</td>
<td></td>
</tr>
<tr>
<td>Step 2 map-name user-attribute-name Cisco-attribute-name</td>
<td>Maps the user-defined attribute name department to the Cisco attribute.</td>
</tr>
<tr>
<td>Example: hostname(config-ldap-attribute-map)# map-name department IETF-Radius-Class</td>
<td></td>
</tr>
<tr>
<td>Step 3 map-value user-attribute-name Cisco-attribute-name</td>
<td>Maps the user-defined map value department to the user-defined attribute value and the Cisco attribute value.</td>
</tr>
<tr>
<td>Example: hostname(config-ldap-attribute-map)# map-value department Engineering group1</td>
<td></td>
</tr>
</tbody>
</table>
### Configuring AAA

#### Examples

The following example shows how to limit management sessions to the ASASM based on an LDAP attribute called `accessType`. The `accessType` attribute has three possible values:

- VPN
- admin
- helpdesk

The following example shows how each value is mapped to one of the valid IETF-Radius-Service-Type attributes that the ASASM supports: remote-access (Service-Type 5) Outbound, admin (Service-Type 6) Administrative, and nas-prompt (Service-Type 7) NAS Prompt:

```
hostname(config)# ldap attribute-map MGMT
hostname(config-ldap-attribute-map)# map-name accessType IETF-Radius-Service-Type
hostname(config-ldap-attribute-map)# map-value accessType VPN 5
hostname(config-ldap-attribute-map)# map-value accessType admin 6
hostname(config-ldap-attribute-map)# map-value accessType helpdesk 7
```

The following example shows how to display the complete list of Cisco LDAP attribute names:

```
hostname(config)# ldap attribute-map att_map_1
hostname(config-ldap-attribute-map)# map-name att_map_1?
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 4 aaa-server server_group [interface_name] host server_ip</td>
<td>Identifies the server and the AAA server group to which it belongs.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# aaa-server ldap_dir_1 host 10.1.1.4</td>
<td></td>
</tr>
<tr>
<td>Step 5 ldap-attribute-map map-name</td>
<td>Binds the attribute map to the LDAP server.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config-aaa-server-host)# ldap-attribute-map att_map_1</td>
<td></td>
</tr>
</tbody>
</table>

Command Purpose

Step 4

**aaa-server server_group [interface_name] host server_ip**

Identifies the server and the AAA server group to which it belongs.

Example:

```
hostname(config)# aaa-server ldap_dir_1 host 10.1.1.4
```

Step 5

**ldap-attribute-map map-name**

Binds the attribute map to the LDAP server.

Example:

```
hostname(config-aaa-server-host)# ldap-attribute-map att_map_1
```
Adding a User Account to the Local Database

This section describes how to manage users in the local database and includes the following topics:

Guidelines

The local database is used for the following features:

- ASDM per-user access
- Console authentication
- Telnet and SSH authentication.
- `enable` command authentication
  This setting is for CLI-access only and does not affect the ASDM login.
- Command authorization
  If you turn on command authorization using the local database, then the ASASM refers to the user privilege level to determine which commands are available. Otherwise, the privilege level is not generally used. By default, all commands are either privilege level 0 or level 15.
- Network access authentication

For multiple context mode, you can configure usernames in the system execution space to provide individual logins at the CLI using the `login` command; however, you cannot configure any AAA rules that use the local database in the system execution space.

Limitations

You cannot use the local database for network access authorization.
To add a user to the local database, perform the following steps:

**Detailed Steps**
Configuring AAA

### Command

**Step 1**

`username username {nopassword | password password } [privilege priv_level]`

**Example:**

```
hostname(config)# username exampleuser1 privilege 1
```

**Purpose**

Creates the user account. The `username username` keyword is a string from 4 to 64 characters long.

**Note**

The ASASM does not prohibit the creation of usernames that only differ by case with previously configured usernames. We do not recommend this practice if VPN users are authenticated using the local user database. Usernames such as “User1” and “user1” are still distinct for authentication purposes, but if a maximum simultaneous login limit has been configured, these users share the same session count. This makes it possible for “user1” to log off “User1” by establishing a tunnel that exceeds the simultaneous login limit.

The `password password` argument is a string from 3 to 32 characters long. You can override the default password by entering the `aaa authentication telnet console` command when you session to the ASASM from the switch. The `privilege level` argument sets the privilege level, which ranges from 0 to 15. The default is 2. This privilege level is used with command authorization.

**Caution**

If you do not use command authorization (the `aaa authorization console LOCAL` command), then the default level 2 allows management access to privileged EXEC mode. To limit access to privileged EXEC mode, either set the privilege level to 0 or 1, or use the `service-type` command (see Step 5).

The `nopassword` keyword creates a user account with no password.

The `encrypted` and `nt-encrypted` keywords are typically for display only. When you define a password in the `username` command, the ASASM encrypts it when it saves it to the configuration for security purposes. When you enter the `show running-config` command, the `username` command does not show the actual password; it shows the encrypted password followed by the `encrypted` or `nt-encrypted` keyword (when you specify `mschap`). For example, if you enter the password “test,” the `show running-config` output would appear as something similar to the following:

```
username user1 password DLaUiAX3i78qgoB5c7iVNw==
nt-encrypted
```

The only time you would actually enter the `encrypted` or `nt-encrypted` keyword at the CLI is if you are cutting and pasting a configuration file for use in another ASASM, and you are using the same password.
Configuring AAA

### Examples

The following example assigns a privilege level of 15 to the admin user account:

```
hostname(config)# username admin password password privilege 15
```

The following example creates a user account with no password:

```
hostname(config)# username user34 nopassword
```
The following example enables management authorization, creates a user account with a password, enters username attributes configuration mode, and specifies the `service-type` attribute:

```plaintext
hostname(config)# aaa authorization exec authentication-server
hostname(config)# username user1 password gOgeOus
hostname(config)# username user1 attributes
hostname(config-username)# service-type nas-prompt
```

### Managing User Passwords

The ASASM enables administrators with the necessary privileges to modify password policy for users in the current context.

User passwords have the following guidelines:

- A maximum lifetime of 0 to 65536 days.
- A minimum length of 3 to 64 characters.
- A minimum number of changed characters for updates of 0 to 64 characters.
- They may include lower case characters.
- They may include upper case characters.
- They may include numbers.
- They may include special characters.

To specify password policy for users, perform the following steps:

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td><code>password-policy lifetime value</code></td>
<td>Sets the password policy for the current context and the interval in days after which passwords expire. Valid values are between 0 and 65536 days. The default value is 0 days.</td>
</tr>
<tr>
<td>Example:</td>
<td><code>hostname (config)# password-policy lifetime 1000</code></td>
<td></td>
</tr>
<tr>
<td>Step 2</td>
<td><code>password-policy minimum-changes value</code></td>
<td>Sets the minimum number of characters that must be changed between new and old passwords. Valid values are between 0 and 64 characters. The default value is 0. New passwords must include a minimum of 4 character changes from the current password and are considered changed only if they do not appear anywhere in the current password.</td>
</tr>
<tr>
<td>Example:</td>
<td><code>hostname(config)# password-policy minimum-changes 4</code></td>
<td></td>
</tr>
<tr>
<td>Step 3</td>
<td><code>password-policy minimum-length value</code></td>
<td>Sets the minimum length of passwords. Valid values are between 3 and 64 characters. The recommended minimum password length is 8 characters. If the minimum length is less than the value of any of the other minimum values (lowercase, numeric, special, and uppercase), an error message appears and the minimum length is not changed.</td>
</tr>
<tr>
<td>Example:</td>
<td><code>hostname(config)# password-policy minimum-length 8</code></td>
<td></td>
</tr>
</tbody>
</table>
### Configuring AAA

#### Changing User Passwords

The ASASM enables administrators with the necessary privileges to modify passwords for users in the current context. Users must authenticate with their current passwords before they are allowed to change passwords. However, authentication is not required when an administrator is changing a user password.

To enable users to change their own account passwords, enter the following command:

**Command**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>change-password [old-password] old-password [new-password] new-password</td>
<td>Enables users to change their own account passwords. The new-password new-password keyword-argument pair specifies the new password. The old-password old-password keyword-argument pair specifies the old password, which reauthenticates the user. If users omit the passwords, the ASA prompts them for input. When users enter the change-password command, they are asked to save their running configuration.</td>
</tr>
</tbody>
</table>

**Example:**

```
hostname# change-password old-password myoldpassword000 new password mynewpassword123
```
### Authenticating Users with a Public Key for SSH

Users can authenticate with a public key for SSH. The public key can be hashed or not hashed.  
To authenticate with a public key for SSH, enter the following command:

```
username {user} attributes ssh authentication publickey key [hashed]
```

**Example:**
```
hostname(config)# username anyuser ssh authentication publickey key [hashed]
```

Enables public key authentication on a per-user basis. The value of the *key* argument can be one of the following:

- When the *key* argument is supplied and the hashed tag is not specified, the value of the key must be a Base 64 encoded public key that is generated by SSH key generation software that can generate SSH-RSA raw keys (that is, with no certificates). After you submit the Base 64 encoded public key, that key is then hashed via SHA-256 and the corresponding 32-byte hash is used for all further comparisons.
- When the *key* argument is supplied and the hashed tag is specified, the value of the key must have been previously hashed with SHA-256 and be 32 bytes long, with each byte separated by a colon (for parsing purposes).

When you save the configuration, the hashed key value is saved to the configuration and used when the ASASM is rebooted.

### Differentiating User Roles Using AAA

The ASASM enables you to distinguish between administrative and remote-access users when they authenticate using RADIUS, LDAP, TACACS+, or the local user database. User role differentiation can prevent remote access VPN and network access users from establishing an administrative connection to the ASASM.

To differentiate user roles, use the `service-type` attribute in `username` configuration mode. For RADIUS and LDAP (with the `ldap-attribute-map` command), you can use a Cisco Vendor-Specific Attribute (VSA), Cisco-Priv-Level, to assign a privilege level to an authenticated user.

This section includes the following topics:

- Using Local Authentication, page 33-25
- Using RADIUS Authentication, page 33-26
- Using LDAP Authentication, page 33-26
- Using TACACS+ Authentication, page 33-27

### Using Local Authentication

Before you configure the `service-type` attribute and privilege level when using local authentication, you must create a user, assign a password, and assign a privilege level.

To do so, enter the following command:

```
hostname(config)# username admin password mysecret123 privilege 15
```
Where mysecret123 is the stored password and 15 is the assigned privilege level, which indicates an admin user.

The available configuration options for the service-type attribute include the following:

- **admin**, in which users are allowed access to the configuration mode. This option also allows a user to connect via remote access.
- **nas-prompt**, in which users are allowed access to the EXEC mode.
- **remote-access**, in which users are allowed access to the network.

The following example designates a service-type of admin for a user named admin:

```
hostname(config)# username admin attributes
hostname(config-username)# service-type admin
```

The following example designates a service-type of remote-access for a user named ra-user:

```
hostname(config)# username ra-user attributes
hostname(config-username)# service-type remote-access
```

### Using RADIUS Authentication

The RADIUS IETF service-type attribute, when sent in an access-accept message as the result of a RADIUS authentication and authorization request, is used to designate which type of service is granted to the authenticated user. The supported attribute values are the following: administrative(6), nas-prompt(7), Framed(2), and Login(1). For a list of supported RADIUS IETF VSAs used for authentication and authorization, see Table C-8 on page C-36.

For more information about using RADIUS authentication, see “Configuring an External RADIUS Server” section on page C-27. For more information about configuring RADIUS authentication for Cisco Secure ACS, see the Cisco Secure ACS documentation on Cisco.com.

The RADIUS Cisco VSA privilege-level attribute (Vendor ID 3076, sub-ID 220), when sent in an access-accept message, is used to designate the level of privilege for the user. For a list of supported RADIUS VSAs used for authorization, see Table C-7 on page C-28.

### Using LDAP Authentication

When users are authenticated through LDAP, the native LDAP attributes and their values can be mapped to Cisco ASASM attributes to provide specific authorization features. For the supported list of LDAP VSAs used for authorization, see Table C-2 on page C-6.

You can use the LDAP attribute mapping feature for LDAP authorization. For examples of this feature, see the “Understanding Policy Enforcement of Permissions and Attributes” section on page C-1.

The following example shows how to define an LDAP attribute map. In this example, the security policy specifies that users being authenticated through LDAP map the user record fields or parameters title and company to the IETF-RADIUS service-type and privilege-level, respectively.

To define an LDAP attribute map, enter the following commands:

```
hostname(config)# ldap attribute-map admin-control
hostname(config-ldap-attribute-map)# map-name title IETF-RADIUS-Service-Type
hostname(config-ldap-attribute-map)# map-name company Privilege-Level
```

The following is sample output from the ldap-attribute-map command:

```
ldap attribute-map admin-control
```
To apply the LDAP attribute map to the LDAP AAA server, enter the following commands:

```
hostname(config)# aaa-server ldap-server (dmz1) host 10.20.30.1
hostname(config-aaa-server-host)# ldap-attribute-map admin-control
```

**Note**

When an authenticated user tries administrative access to the ASASM through ASDM, SSH, or Telnet, but does not have the appropriate privilege level to do so, the ASASM generates syslog message 113021. This message informs the user that the attempted login failed because of inappropriate administrative privileges.

### Using TACACS+ Authentication

For information about how to configure TACACS+ authentication, see the “RADIUS Accounting Disconnect Reason Codes” section on page C-37.

### Monitoring AAA Servers

To monitor AAA servers, enter one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| `show aaa-server` | Shows the configured AAA server statistics.  
To clear the AAA server configuration, enter the `clear aaa-server statistics` command. |
| `show running-config aaa-server` | Shows the AAA server running configuration.  
To clear AAA server statistics, enter the `clear configure aaa-server` command. |
| `show running-config all ldap attribute-map` | Shows all LDAP attribute maps in the running configuration.  
To clear all LDAP attribute maps in the running configuration, use the `clear configuration ldap attribute-map` command. |
| `show running-config zonelabs-integrity` | Shows the Zone Labs Integrity server configuration.  
To clear the Zone Labs Integrity server configuration, use the `clear configure zonelabs-integrity` command. |
| `show ad-groups name [filter string]` | Applies only to AD servers using LDAP, and shows groups that are listed on an AD server. |
| `show running-config [all] password-policy` | Shows the password policy for the current context. |
Additional References

For additional information related to implementing LDAP mapping, see the “RFCs” section on page 33-28.

RFCs

<table>
<thead>
<tr>
<th>RFC</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>2138</td>
<td>Remote Authentication Dial In User Service (RADIUS)</td>
</tr>
<tr>
<td>2139</td>
<td>RADIUS Accounting</td>
</tr>
<tr>
<td>2548</td>
<td>Microsoft Vendor-specific RADIUS Attributes</td>
</tr>
<tr>
<td>2868</td>
<td>RADIUS Attributes for Tunnel Protocol Support</td>
</tr>
</tbody>
</table>

Feature History for AAA Servers

Table 33-3 lists each feature change and the platform release in which it was implemented.
Table 33-3 Feature History for AAA Servers

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>AAA Servers</td>
<td>7.0(1)</td>
<td>AAA Servers describe support for AAA and how to configure AAA servers and the local database. We introduced the following commands:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>username, aaa authorization exec authentication-server, aaa authentication console LOCAL, aaa authorization exec LOCAL, service-type, ldap attribute-map, aaa-server protocol, aaa authentication {telnet</td>
</tr>
<tr>
<td>Key vendor-specific attributes (VSAs) sent in RADIUS access request and accounting request packets from the ASASM</td>
<td>8.4(3)</td>
<td>Four New VSAs—Tunnel Group Name (146) and Client Type (150) are sent in RADIUS access request packets from the ASASM. Session Type (151) and Session Subtype (152) are sent in RADIUS accounting request packets from the ASASM. All four attributes are sent for all accounting request packet types: Start, Interim-Update, and Stop. The RADIUS server (for example, ACS and ISE) can then enforce authorization and policy attributes or use them for accounting and billing purposes.</td>
</tr>
<tr>
<td>Common Criteria certification and FIPS support for password policy, password change, and SSH public key authentication</td>
<td>8.4(4.1)</td>
<td>We introduced or modified the following commands: password-policy lifetime, password-policy minimum changes, password-policy minimum-length, password-policy minimum-lowercase, password-policy minimum-uppercase, password-policy minimum-numeric, password-policy minimum-special, password-policy authenticate enable, username, username attributes, clear configure username, change-password, clear configure password-policy, show running-config password-policy, and username.</td>
</tr>
</tbody>
</table>
Configuring Management Access

This chapter describes how to access the ASASM for system management through Telnet, SSH, and HTTPS (using ASDM), how to authenticate and authorize users, how to create login banners, and how to customize CLI parameters.

This chapter includes the following sections:

- Configuring ASA Access for ASDM, Telnet, or SSH, page 34-1
- Configuring CLI Parameters, page 34-6
- Configuring ICMP Access, page 34-10
- Configuring AAA for System Administrators, page 34-12
- Feature History for Management Access, page 34-32

To access the ASASM interface for management access, you do not also need an access list allowing the host IP address. You only need to configure management access according to the sections in this chapter.

Configuring ASA Access for ASDM, Telnet, or SSH

This section describes how to allow clients to access the ASASM using ASDM, Telnet, or SSH and includes the following topics:

- Licensing Requirements for ASA Access for ASDM, Telnet, or SSH, page 34-1
- Guidelines and Limitations, page 34-2
- Configuring Telnet Access, page 34-3
- Using a Telnet Client, page 34-4
- Configuring SSH Access, page 34-4
- Using an SSH Client, page 34-5
- Configuring HTTPS Access for ASDM, page 34-6

Licensing Requirements for ASA Access for ASDM, Telnet, or SSH

The following table shows the licensing requirements for this feature:
Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

**Context Mode Guidelines**
Supported in single and multiple context mode.

**Firewall Mode Guidelines**
Supported in routed and transparent firewall mode.

**IPv6 Guidelines**
Supports IPv6.

**Model Guidelines**
For the ASASM, a session from the switch to the ASASM is a Telnet session, but Telnet access configuration according to this section is not required.

**Additional Guidelines**
- You cannot use Telnet to the lowest security interface.
- Management access to an interface other than the one from which you entered the ASASM is not supported. For example, if your management host is located on the outside interface, you can only initiate a management connection directly to the outside interface.
- The ASASM allows:
  - A maximum of 5 concurrent Telnet connections per context, if available, with a maximum of 100 connections divided among all contexts.
  - A maximum of 5 concurrent SSH connections per context, if available, with a maximum of 100 connections divided among all contexts.
  - A maximum of 5 concurrent ASDM instances per context, if available, with a maximum of 32 ASDM instances among all contexts.
- The ASASM supports the SSH remote shell functionality provided in SSH Versions 1 and 2 and supports DES and 3DES ciphers.
- XML management over SSL and SSH is not supported.
- (8.4 and later) The SSH default username is no longer supported. You can no longer connect to the ASA using SSH with the **pix** or **asa** username and the login password. To use SSH, you must configure AAA authentication using the **aaa authentication ssh console LOCAL** command; then define a local user by entering the **username** command. If you want to use a AAA server for authentication instead of the local database, we recommend also configuring local authentication as a backup method.
Configuring Telnet Access

To identify the client IP addresses allowed to connect to the ASASM using Telnet, perform the following steps.

### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>telnet source_IP_address mask source_interface</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname(config)# telnet 192.168.1.2 255.255.255.255 inside</td>
</tr>
<tr>
<td>For each address or subnet, identifies the IP addresses from which the ASASM accepts connections.</td>
<td></td>
</tr>
<tr>
<td>If there is only one interface, you can configure Telnet to access that interface as long as the interface has a security level of 100.</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>telnet timeout minutes</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname(config)# telnet timeout 30</td>
</tr>
<tr>
<td>Sets the duration for how long a Telnet session can be idle before the ASASM disconnects the session.</td>
<td></td>
</tr>
<tr>
<td>Set the timeout from 1 to 1440 minutes. The default is 5 minutes. The default duration is too short in most cases and should be increased until all pre-production testing and troubleshooting have been completed.</td>
<td></td>
</tr>
</tbody>
</table>

### Examples

The following example shows how to let a host on the inside interface with an address of 192.168.1.2 access the ASASM:

hostname(config)# telnet 192.168.1.2 255.255.255.255 inside

The following example shows how to allow all users on the 192.168.3.0 network to access the ASASM on the inside interface:

hostname(config)# telnet 192.168.3.0 255.255.255.0 inside
Using a Telnet Client

To gain access to the ASASM CLI using Telnet, enter the login password set by the `password` command. If you configure Telnet authentication (see the “Configuring Authentication for CLI and ASDM Access” section on page 34-18), then enter the username and password defined by the AAA server or local database.

Configuring SSH Access

To identify the client IP addresses and define a user allowed to connect to the ASASM using SSH, perform the following steps.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 crypto key generate rsa modulus</td>
<td>Generates an RSA key pair, which is required for SSH. The modulus value (in bits) is 512, 768, 1024, or 2048. The larger the key modulus size you specify, the longer it takes to generate an RSA key pair. We recommend a value of 1024.</td>
</tr>
<tr>
<td>modulus_size</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# crypto key generate rsa modulus 1024</td>
<td></td>
</tr>
<tr>
<td>Step 2 write memory</td>
<td>Saves the RSA keys to persistent flash memory.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# write memory</td>
<td></td>
</tr>
<tr>
<td>Step 3 aaa authentication ssh console LOCAL</td>
<td>Enables local authentication for SSH access. You can alternatively configure authentication using a AAA server. See the “Configuring Authentication for CLI and ASDM Access” section on page 34-18 for more information.</td>
</tr>
<tr>
<td>Step 4 username username password password</td>
<td>Creates a user in the local database that can be used for SSH access.</td>
</tr>
<tr>
<td>Step 5 ssh source_IP_address mask source_interface</td>
<td>For each address or subnet, identifies the IP addresses from which the ASASM accepts connections, and the interface on which you can SSH. Unlike Telnet, you can SSH on the lowest security level interface.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# ssh 192.168.3.0 255.255.255.0 inside</td>
<td></td>
</tr>
<tr>
<td>Step 6 (Optional) ssh timeout minutes</td>
<td>Sets the duration for how long an SSH session can be idle before the ASASM disconnects the session. Set the timeout from 1 to 60 minutes. The default is 5 minutes. The default duration is too short in most cases, and should be increased until all pre-production testing and troubleshooting have been completed.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# ssh timeout 30</td>
<td></td>
</tr>
</tbody>
</table>
### Examples

The following example shows how to generate RSA keys and let a host on the inside interface with an address of 192.168.1.2 access the ASASM:

```
hostname(config)# crypto key generate rsa modulus 1024
hostname(config)# write memory
hostname(config)# aaa authentication ssh console LOCAL
WARNING: local database is empty! Use 'username' command to define local users.
hostname(config)# username exampleuser1 password examplepassword1
hostname(config)# ssh 192.168.1.2 255.255.255.255 inside
hostname(config)# ssh timeout 30
```

The following example shows how to allow all users on the 192.168.3.0 network to access the ASASM on the inside interface:

```
hostname(config)# ssh 192.168.3.0 255.255.255.0 inside
```

### Using an SSH Client

In the SSH client on your management host, enter the username and password that you configured in the “Configuring SSH Access” section on page 34-4. When starting an SSH session, a dot (.) displays on the ASASM console before the following SSH user authentication prompt appears:

```
hostname(config)# .
```

The display of the dot does not affect the functionality of SSH. The dot appears at the console when generating a server key or decrypting a message using private keys during SSH key exchange before user authentication occurs. These tasks can take up to two minutes or longer. The dot is a progress indicator that verifies that the ASASM is busy and has not hung.

**Note**

If more than one SSH configuration session exists and the configuration operation is carried through any file operations (such as copy, tftp, config net, context mode config file), even if it is a single CLI, it will be blocked with the response "Command Ignored, configuration in progress...". If the CLI is directly entered through a command prompt, it is not blocked.
Configuring HTTPS Access for ASDM

To use ASDM, you need to enable the HTTPS server, and allow HTTPS connections to the ASASM. HTTPS access is enabled as part of the factory default configuration or when you use the `setup` command. This section describes how to manually configure ASDM access.

To configure HTTPS access for ASDM, perform the following steps:

### Detailed Steps

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><code>http source_IP_address mask source_interface</code></td>
<td>For each address or subnet, identifies the IP addresses from which the ASASM accepts HTTPS connections.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>hostname(config)# http 192.168.1.2 255.255.255.255 inside</code></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td><code>http server enable [port]</code></td>
<td>Enables the HTTPS server.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>hostname(config)# http server enable 443</code></td>
<td></td>
</tr>
</tbody>
</table>

### Examples

The following example shows how to enable the HTTPS server and let a host on the inside interface with an address of 192.168.1.2 access ASDM:

```
hostname(config)# http server enable
hostname(config)# http 192.168.1.2 255.255.255.255 inside
```

The following example shows how to allow all users on the 192.168.3.0 network to access ASDM on the inside interface:

```
hostname(config)# http 192.168.3.0 255.255.255.255 inside
```

### Configuring CLI Parameters

This section includes the following topics:

- Licensing Requirements for CLI Parameters, page 34-7
- Guidelines and Limitations, page 34-7
- Configuring a Login Banner, page 34-7
- Customizing a CLI Prompt, page 34-8
- Changing the Console Timeout, page 34-9
Licensing Requirements for CLI Parameters

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

**Context Mode Guidelines**
Supported in single and multiple context mode.

**Firewall Mode Guidelines**
Supported in routed and transparent firewall mode.

Configuring a Login Banner

You can configure a message to display when a user connects to the ASASM, before a user logs in, or before a user enters privileged EXEC mode.

**Restrictions**

After a banner is added, Telnet or SSH sessions to ASASM may close if:

- There is not enough system memory available to process the banner message(s).
- A TCP write error occurs when trying to display banner message(s).

**Guidelines**

- From a security perspective, it is important that your banner discourage unauthorized access. Do not use the words “welcome” or “please,” as they appear to invite intruders in. The following banner sets the correct tone for unauthorized access:
  
  You have logged in to a secure device. If you are not authorized to access this device, log out immediately or risk possible criminal consequences.

- See RFC 2196 for guidelines about banner messages.
To configure a login banner, perform the following steps:

**Detailed Steps**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>banner (exec</td>
<td>login</td>
</tr>
</tbody>
</table>

- Spaces are allowed, but tabs cannot be entered using the CLI.
- There are no limits for banner length other than those for RAM and flash memory.
- You can dynamically add the hostname or domain name of the ASASM by including the strings $(hostname) and $(domain).
- If you configure a banner in the system configuration, you can use that banner text within a context by using the $(system) string in the context configuration. |

**Examples**

The following example shows how to add a message-of-the-day banner:

```
hostname(config)# banner motd Welcome to $(hostname).
hostname(config)# banner motd Contact me at admin@example.com for any issues.
```

**Customizing a CLI Prompt**

The CLI Prompt pane lets you customize the prompt used during CLI sessions. By default, the prompt shows the hostname of the ASASM. In multiple context mode, the prompt also displays the context name. You can display the following items in the CLI prompt:

<table>
<thead>
<tr>
<th>context</th>
<th>(Multiple mode only) Displays the name of the current context.</th>
</tr>
</thead>
<tbody>
<tr>
<td>domain</td>
<td>Displays the domain name.</td>
</tr>
<tr>
<td>hostname</td>
<td>Displays the hostname.</td>
</tr>
</tbody>
</table>
Chapter 34      Configuring Management Access

Configuring CLI Parameters

Detailed Steps

To customize the CLI prompt, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>prompt {hostname} {context} {domain} {slot} {state} {priority} }</td>
<td>Customizes the CLI prompt.</td>
</tr>
</tbody>
</table>

Example:
hostname(config)# firewall transparent

Changing the Console Timeout

The console timeout sets how long a connection can remain in privileged EXEC mode or configuration mode; when the timeout is reached, the session drops into user EXEC mode. By default, the session does not time out. This setting does not affect how long you can remain connected to the console port, which never times out.

To change the console timeout, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>console timeout number</td>
<td>Specifies the idle time in minutes (0 through 60) after which the privileged session ends. The default timeout is 0, which means the session does not time out.</td>
</tr>
</tbody>
</table>

Example:
hostname(config)# console timeout 0

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>
Configuring ICMP Access

By default, you can send ICMP packets to any ASASM interface using either IPv4 or IPv6. This section tells how to limit ICMP management access to the ASASM. You can protect the ASASM from attacks by limiting the addresses of hosts and networks that are allowed to have ICMP access to the ASASM.

For allowing ICMP traffic through the ASASM, see Chapter 32, “Configuring Access Rules.”

This section includes the following topics:
- Information About ICMP Access, page 34-10
- Licensing Requirements for ICMP Access, page 34-10
- Guidelines and Limitations, page 34-10
- Default Settings, page 34-11
- Configuring ICMP Access, page 34-11

Information About ICMP Access

ICMP in IPv6 functions the same as ICMP in IPv4. ICMPv6 generates error messages, such as ICMP destination unreachable messages and informational messages like ICMP echo request and reply messages. Additionally ICMP packets in IPv6 are used in the IPv6 neighbor discovery process and path MTU discovery.

We recommend that you always grant permission for the ICMP unreachable message type (type 3). Denying ICMP unreachable messages disables ICMP path MTU discovery, which can halt IPsec and PPTP traffic. See RFC 1195 and RFC 1435 for details about path MTU discovery.

If you configure ICMP rules, then the ASASM uses a first match to the ICMP traffic followed by an implicit deny all entry. That is, if the first matched entry is a permit entry, the ICMP packet continues to be processed. If the first matched entry is a deny entry or an entry is not matched, the ASASM discards the ICMP packet and generates a syslog message. An exception is when an ICMP rule is not configured; in that case, a permit statement is assumed.

Licensing Requirements for ICMP Access

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

Context Mode Guidelines

Supported in single and multiple context mode.
Firewall Mode Guidelines
Supported in routed and transparent firewall mode.

IPv6 Guidelines
Supports IPv6.

Additional Guidelines
- The ASASM does not respond to ICMP echo requests directed to a broadcast address.
- The ASASM only responds to ICMP traffic sent to the interface that traffic comes in on; you cannot send ICMP traffic through an interface to a far interface.

Default Settings
By default, you can send ICMP packets to any ASASM interface using either IPv4 or IPv6.

Configuring ICMP Access
To configure ICMP access rules, enter one of the following commands:

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>(For IPv4)</td>
<td></td>
</tr>
<tr>
<td>icmp (permit</td>
<td>deny) {host ip_address</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# icmp deny host 10.1.1.15 inside</td>
<td></td>
</tr>
<tr>
<td>(For IPv6)</td>
<td></td>
</tr>
<tr>
<td>ipv6 icmp (permit</td>
<td>deny) {ipv6-prefix/prefix-length</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# icmp permit host fe80::20d:88ff:feee:6a82 outside</td>
<td></td>
</tr>
</tbody>
</table>

Examples
The following example shows how to allow all hosts except the one at 10.1.1.15 to use ICMP to the inside interface:

hostname(config)# icmp deny host 10.1.1.15 inside
hostname(config)# icmp permit any inside
The following example shows how to allow the host at 10.1.1.15 to use only ping to the inside interface, enter the following command:

```
hostname(config)# icmp permit host 10.1.1.15 inside
```

The following example shows how to deny all ping requests and permit all packet-too-big messages (to support path MTU discovery) at the outside interface:

```
hostname(config)# ipv6 icmp deny any echo-reply outside
hostname(config)# ipv6 icmp permit any packet-too-big outside
```

The following example shows how to permit host 2000:0:4::2 or hosts on prefix 2001::/64 to ping the outside interface:

```
hostname(config)# ipv6 icmp permit host 2000:0:4::2 echo-reply outside
hostname(config)# ipv6 icmp permit 2001::/64 echo-reply outside
hostname(config)# ipv6 icmp permit any packet-too-big outside
```

### Configuring AAA for System Administrators

This section describes how to enable authentication and command authorization for system administrators. Before you configure AAA for system administrators, first configure the local database or AAA server according to procedures listed in Chapter 33, “Configuring AAA Servers and the Local Database.”

This section includes the following topics:

- Information About AAA for System Administrators, page 34-13
- Licensing Requirements for AAA for System Administrators, page 34-16
- Prerequisites, page 34-16
- Guidelines and Limitations, page 34-17
- Default Settings, page 34-17
- Configuring Authentication for CLI and ASDM Access, page 34-18
- For the ASASM, this parameter affects the virtual console accessed from the switch using the service-module session command. For multiple mode access, see the Configuring Authentication to Access Privileged EXEC Mode (the enable Command), page 34-19
- Limiting User CLI and ASDM Access with Management Authorization, page 34-20
- Configuring Command Authorization, page 34-21
- Configuring Management Access Accounting, page 34-29
- Viewing the Currently Logged-In User, page 34-29
- Recovering from a Lockout, page 34-30

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>
Information About AAA for System Administrators

This section describes AAA for system administrators and includes the following topics:

- Information About Management Authentication, page 34-13
- Information About Command Authorization, page 34-14

Information About Management Authentication

This section describes authentication for management access and includes the following topics:

- Comparing CLI Access with and without Authentication, page 34-13
- Comparing ASDM Access with and without Authentication, page 34-14
- Authenticating Sessions from the Switch to the ASA Services Module, page 34-14

Comparing CLI Access with and without Authentication

How you log into the ASASM depends on whether or not you enable authentication:

- If you do not enable any authentication for Telnet, you do not enter a username; you enter the login password (set with the `password` command). For SSH, you enter the username and the login password. You access user EXEC mode.

- If you enable Telnet or SSH authentication according to this section, you enter the username and password as defined on the AAA server or local user database. You access user EXEC mode.

To enter privileged EXEC mode after logging in, enter the `enable` command. How `enable` works depends on whether you enable authentication:

- If you do not configure enable authentication, enter the system enable password when you enter the `enable` command (set by the `enable password` command). However, if you do not use enable authentication, after you enter the `enable` command, you are no longer logged in as a particular user. To maintain your username, use enable authentication.

- If you configure enable authentication (see the For the ASASM, this parameter affects the virtual console accessed from the switch using the service-module session command. For multiple mode access, see the For the ASASM, this parameter also affects the session from the switch using the session command. For multiple mode access, see the Configuring Authentication to Access Privileged EXEC Mode (the enable Command), page 34-19), the ASASM prompts you for your username and password again. This feature is particularly useful when you perform command authorization, in which usernames are important in determining the commands that a user can enter.

For enable authentication using the local database, you can use the `login` command instead of the `enable` command. `login` maintains the username but requires no configuration to turn on authentication. See the “Authenticating Users with the login Command” section on page 34-19 for more information.
Comparing ASDM Access with and without Authentication

By default, you can log into ASDM with a blank username and the enable password set by the `enable password` command. Note that if you enter a username and password at the login screen (instead of leaving the username blank), ASDM checks the local database for a match.

If you configure HTTP authentication, you can no longer use ASDM with a blank username and the enable password.

Authenticating Sessions from the Switch to the ASA Services Module

For sessions from the switch to the ASASM (using the `session` command), you can configure Telnet authentication. For virtual console connections from the switch to the ASASM (using the `service-module session` command), you can configure serial port authentication.

In multiple context mode, you cannot configure any AAA commands in the system configuration. However, if you configure Telnet or serial authentication in the admin context, then authentication also applies to sessions from the switch to the ASASM. The admin context AAA server or local user database is used in this instance.

Information About Command Authorization

This section describes command authorization and includes the following topics:

- Supported Command Authorization Methods, page 34-14
- About Preserving User Credentials, page 34-15
- Security Contexts and Command Authorization, page 34-15

Supported Command Authorization Methods

You can use one of two command authorization methods:

- Local privilege levels—Configure the command privilege levels on the ASASM. When a local, RADIUS, or LDAP (if you map LDAP attributes to RADIUS attributes) user authenticates for CLI access, the ASASM places that user in the privilege level that is defined by the local database, RADIUS, or LDAP server. The user can access commands at the assigned privilege level and below. Note that all users access user EXEC mode when they first log in (commands at level 0 or 1). The user needs to authenticate again with the `enable` command to access privileged EXEC mode (commands at level 2 or higher), or they can log in with the `login` command (local database only).

  **Note** You can use local command authorization without any users in the local database and without CLI or `enable` authentication. Instead, when you enter the `enable` command, you enter the system enable password, and the ASASM places you in level 15. You can then create enable passwords for every level, so that when you enter `enable n` (2 to 15), the ASASM places you in level n. These levels are not used unless you enable local command authorization (see the “Configuring Local Command Authorization” section on page 34-22). (See the command reference for more information about the `enable` command.)

- TACACS+ server privilege levels—On the TACACS+ server, configure the commands that a user or group can use after authenticating for CLI access. Every command that a user enters at the CLI is validated with the TACACS+ server.
About Preserving User Credentials

When a user logs into the ASASM, that user is required to provide a username and password for authentication. The ASASM retains these session credentials in case further authentication is needed later in the session.

When the following configurations are in place, a user needs only to authenticate with the local server for login. Subsequent serial authorization uses the saved credentials. The user is also prompted for the privilege level 15 password. When exiting privileged mode, the user is authenticated again. User credentials are not retained in privileged mode.

- The local server is configured to authenticate user access.
- Privilege level 15 command access is configured to require a password.
- The user account is configured for serial-only authorization (no access to console or ASDM).
- The user account is configured for privilege level 15 command access.

The following table shows how credentials are used in this case by the ASASM.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Username</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Password</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Privileged Mode</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Password</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>

Security Contexts and Command Authorization

The following are important points to consider when implementing command authorization with multiple security contexts:

- AAA settings are discrete per context, not shared among contexts.
  When configuring command authorization, you must configure each security context separately. This configuration provides you the opportunity to enforce different command authorizations for different security contexts.

  When switching between security contexts, administrators should be aware that the commands permitted for the username specified when they login may be different in the new context session or that command authorization may not be configured at all in the new context. Failure to understand that command authorizations may differ between security contexts could confuse an administrator. This behavior is further complicated by the next point.

- New context sessions started with the `changeto` command always use the default enable_15 username as the administrator identity, regardless of which username was used in the previous context session. This behavior can lead to confusion if command authorization is not configured for the enable_15 user or if authorizations are different for the enable_15 user than for the user in the previous context session.

  This behavior also affects command accounting, which is useful only if you can accurately associate each command that is issued with a particular administrator. Because all administrators with permission to use the `changeto` command can use the enable_15 username in other contexts,
command accounting records may not readily identify who was logged in as the enable_15 username. If you use different accounting servers for each context, tracking who was using the enable_15 username requires correlating the data from several servers.

When configuring command authorization, consider the following:

- An administrator with permission to use the `changentto` command effectively has permission to use all commands permitted to the enable_15 user in each of the other contexts.
- If you intend to authorize commands differently per context, ensure that in each context the enable_15 username is denied use of commands that are also denied to administrators who are permitted use of the `changentto` command.

When switching between security contexts, administrators can exit privileged EXEC mode and enter the `enable` command again to use the username that they need.

---

**Note**
The system execution space does not support AAA commands; therefore, command authorization is not available in the system execution space.

### Licensing Requirements for AAA for System Administrators

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

### Prerequisites

Depending on the feature, you can use the following:

- AAA server—See the “Configuring AAA Server Groups” section on page 33-11.
- Local Database—See the “Adding a User Account to the Local Database” section on page 33-19.

**Prerequisites for Management Authentication**

Before the ASASM can authenticate a Telnet, SSH, or HTTP user, you must identify the IP addresses that are allowed to communicate with the ASASM. For the ASASM, the exception is for access to the system in multiple context mode; a session from the switch to the ASASM is a Telnet session, but Telnet access configuration is not required. For more information, see the “Configuring ASA Access for ASDM, Telnet, or SSH” section on page 34-1.

**Prerequisites for Local Command Authorization**

- Configure `enable` authentication. (See the “Configuring Authentication for CLI and ASDM Access” section on page 34-18.) `enable` authentication is essential for maintaining the username after the user accesses the `enable` command.

  Alternatively, you can use the `login` command (which is the same as the `enable` command with authentication; for the local database only), which requires no configuration. We do not recommend this option because it is not as secure as `enable` authentication.

  You can also use CLI authentication, but it is not required.
See the following prerequisites for each user type:

- Local database users—Configure each user in the local database at a privilege level from 0 to 15.
- RADIUS users—Configure the user with Cisco VSA CVVPN3000-Privilege-Level with a value between 0 and 15.
- LDAP users—Configure the user with a privilege level between 0 and 15, and then map the LDAP attribute to Cisco VSA CVVPN3000-Privilege-Level according to the “Configuring LDAP Attribute Maps” section on page 33-16.

**Prerequisites for TACACS+ Command Authorization**

- Configure CLI authentication (see the “Configuring Authentication for CLI and ASDM Access” section on page 34-18).
- Configure `enable` authentication (see the “For the ASASM, this parameter affects the virtual console accessed from the switch using the service-module session command. For multiple mode access, see the. For the ASASM, this parameter also affects the session from the switch using the session command. For multiple mode access, see the. Configuring Authentication to Access Privileged EXEC Mode (the enable Command)” section on page 34-19).

**Prerequisites for Management Accounting**

- Configure CLI authentication (see the “Configuring Authentication for CLI and ASDM Access” section on page 34-18).
- Configure `enable` authentication (see the “For the ASASM, this parameter affects the virtual console accessed from the switch using the service-module session command. For multiple mode access, see the. For the ASASM, this parameter also affects the session from the switch using the session command. For multiple mode access, see the. Configuring Authentication to Access Privileged EXEC Mode (the enable Command)” section on page 34-19).

**Guidelines and Limitations**

This section includes the guidelines and limitations for this feature.

**Context Mode Guidelines**

Supported in single and multiple context mode.

**Firewall Mode Guidelines**

Supported in routed and transparent firewall mode.

**IPv6 Guidelines**

Supports IPv6.

**Default Settings**

By default, the following commands are assigned to privilege level 0. All other commands are assigned to privilege level 15.

- `show checksum`
- `show curpriv`
- `enable`
• help
• show history
• login
• logout
• pager
• show pager
• clear pager
• quit
• show version

If you move any configure mode commands to a lower level than 15, be sure to move the configure command to that level as well, otherwise, the user will not be able to enter configuration mode.

To view all privilege levels, see the “Viewing Local Command Privilege Levels” section on page 34-25.

Configuring Authentication for CLI and ASDM Access

To configure management authentication, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa authentication {telnet</td>
<td>ssh</td>
</tr>
</tbody>
</table>

Example:
hostname(config)# aaa authentication
telnet console LOCAL
For the ASASM, this parameter affects the virtual console accessed from the switch using the `service-module session` command. For multiple mode access, see the `session` command. For multiple mode access, see the. Configuring Authentication to Access Privileged EXEC Mode (the `enable` Command)

You can configure the ASASM to authenticate users with a AAA server or the local database when they enter the `enable` command. Alternatively, users are automatically authenticated with the local database when they enter the `login` command, which also accesses privileged EXEC mode depending on the user level in the local database.

This section includes the following topics:
- Configuring Authentication for the `enable` Command, page 34-19
- Authenticating Users with the `login` Command, page 34-19

### Configuring Authentication for the `enable` Command

You can configure the ASASM to authenticate users when they enter the `enable` command. See the “Comparing CLI Access with and without Authentication” section on page 34-13 for more information. To authenticate users who enter the `enable` command, enter the following command:

```
AAA authentication enable console {LOCAL | server_group [LOCAL]}
```

**Example:**
```
hostname(config)# aaa authentication enable console LOCAL
```

This command authenticates users who enter the `enable` command. The user is prompted for the username and password. If you use a AAA server group for authentication, you can configure the ASASM to use the local database as a fallback method if the AAA server is unavailable. Specify the server group name followed by LOCAL (LOCAL is case sensitive). We recommend that you use the same username and password in the local database as the AAA server, because the ASASM prompt does not give any indication of which method is being used. You can alternatively use the local database as your primary method of authentication (with no fallback) by entering LOCAL alone.

### Authenticating Users with the `login` Command

From user EXEC mode, you can log in as any username in the local database using the `login` command. This feature allows users to log in with their own username and password to access privileged EXEC mode, so you do not have to provide the system enable password to everyone. To allow users to access privileged EXEC mode (and all commands) when they log in, set the user privilege level to 2 (the default) through 15. If you configure local command authorization, then the user can only enter commands assigned to that privilege level or lower. See the “Configuring Local Command Authorization” section on page 34-22 for more information.

**Caution**

If you add users to the local database who can gain access to the CLI and whom you do not want to enter privileged EXEC mode, you should configure command authorization. Without command authorization, users can access privileged EXEC mode (and all commands) at the CLI using their own password if their...
privilege level is 2 or greater (2 is the default). Alternatively, you can use a AAA server for
authentication, or you can set all local users to level 1 so you can control who can use the system enable
password to access privileged EXEC mode.

To log in as a user from the local database, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>login</td>
<td>Logs in as a user from the local database. The ASASM prompts for your</td>
</tr>
<tr>
<td></td>
<td>username and password. After you enter your password, the ASASM places</td>
</tr>
<tr>
<td></td>
<td>you in the privilege level that the local database specifies.</td>
</tr>
</tbody>
</table>

**Example:**

```plaintext
hostname# login
```

---

### Limiting User CLI and ASDM Access with Management Authorization

If you configure CLI or `enable` authentication, you can limit a local user, RADIUS, TACACS+, or LDAP
user (if you map LDAP attributes to RADIUS attributes) from accessing the CLI, ASDM, or the `enable`
command.

**Note**

Serial access is not included in management authorization, so if you configure the `aaa authentication
serial console` command, then any user who authenticates can access the console port.

To limit user CLI and ASDM access, perform the following steps:

**Detailed Steps**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td></td>
</tr>
<tr>
<td>`aaa authorization exec</td>
<td>Enables management authorization for local, RADIUS, LDAP</td>
</tr>
<tr>
<td>authentication-server</td>
<td>(mapped), and TACACS+ users. Also enables support of administrative</td>
</tr>
<tr>
<td></td>
<td>user privilege levels from RADIUS, which can be used in conjunction</td>
</tr>
<tr>
<td></td>
<td>with local command privilege levels for command authorization. See the</td>
</tr>
<tr>
<td></td>
<td>“Configuring Local Command Authorization” section on page 34-22 for</td>
</tr>
<tr>
<td></td>
<td>more information. Use the <code>aaa authorization exec LOCAL</code> command to</td>
</tr>
<tr>
<td></td>
<td>enable attributes to be taken from the local database.</td>
</tr>
</tbody>
</table>

**Example:**

```plaintext
hostname(config)# aaa authorization exec authentication-server
```
Configuring Command Authorization

If you want to control access to commands, the ASASM lets you configure command authorization, where you can determine which commands that are available to a user. By default when you log in, you can access user EXEC mode, which offers only minimal commands. When you enter the enable command (or the login command when you use the local database), you can access privileged EXEC mode and advanced commands, including configuration commands.

You can use one of two command authorization methods:

- Local privilege levels
- TACACS+ server privilege levels

### Step 2

To configure the user for management authorization, see the following requirements for each AAA server type or local user:

- **RADIUS or LDAP (mapped) users**—Use the IETF RADIUS numeric Service-Type attribute, which maps to one of the following values:
  - Service-Type 6 (Administrative)—Allows full access to any services specified by the `aaa authentication console` commands.
  - Service-Type 7 (NAS prompt)—Allows access to the CLI when you configure the `aaa authentication {telnet | ssh} console` command, but denies ASDM configuration access if you configure the `aaa authentication http console` command. ASDM monitoring access is allowed. If you configure `enable` authentication with the `aaa authentication enable console` command, the user cannot access privileged EXEC mode using the `enable` command.
  - Service-Type 5 (Outbound)—Denies management access. The user cannot use any services specified by the `aaa authentication console` commands (excluding the `serial` keyword; serial access is allowed). Remote access (IPsec and SSL) users can still authenticate and terminate their remote access sessions.

Configure Cisco VSA CVPN3000-Privilege-Level with a value between 0 and 15, and then map the LDAP attributes to Cisco VAS CVPN3000-Privilege-Level using the `ldap map-attributes` command. For more information, see the "Configuring LDAP Attribute Maps" section on page 33-16.

- **TACACS+ users**—Authorization is requested with “service=shell,” and the server responds with PASS or FAIL.
  - PASS, privilege level 1—Allows access to ASDM, with limited read-only access to the configuration and monitoring sections, and access for `show` commands that are privilege level 1 only.
  - PASS, privilege level 2 and higher—Allows access to the CLI when you configure the `aaa authentication {telnet | ssh} console` command, but denies ASDM configuration access if you configure the `aaa authentication http console` command. ASDM monitoring access is allowed. If you configure `enable` authentication with the `aaa authentication enable console` command, the user cannot access privileged EXEC mode using the `enable` command. You are not allowed to access privileged EXEC mode using the `enable` command if your enable privilege level is set to 14 or less.
  - FAIL—Denies management access. You cannot use any services specified by the `aaa authentication console` commands (excluding the `serial` keyword; serial access is allowed).

- **Local users**—Sets the `service-type` command. By default, the `service-type` is `admin`, which allows full access to any services specified by the `aaa authentication console` command. Uses the `username` command to configure local database users at a privilege level from 0 to 15. For more information, see the “Adding a User Account to the Local Database” section on page 33-19.
Configuring AAA for System Administrators

For more information about command authorization, see the “Information About Command Authorization” section on page 34-14.

This section includes the following topics:

- Configuring Local Command Authorization, page 34-22
- Viewing Local Command Privilege Levels, page 34-25
- Configuring Commands on the TACACS+ Server, page 34-25
- Configuring TACACS+ Command Authorization, page 34-28

Configuring Local Command Authorization

Local command authorization lets you assign commands to one of 16 privilege levels (0 to 15). By default, each command is assigned either to privilege level 0 or 15. You can define each user to be at a specific privilege level, and each user can enter any command at the assigned privilege level or below. The ASASM supports user privilege levels defined in the local database, a RADIUS server, or an LDAP server (if you map LDAP attributes to RADIUS attributes. See the “Configuring LDAP Attribute Maps” section on page 33-16.)
To configure local command authorization, perform the following steps:

**Detailed Steps**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> privilege [show</td>
<td>clear</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# privilege show level 5 command filter</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 2</strong> aaa authorization exec authentication-server</td>
<td>Supports administrative user privilege levels from RADIUS. Enforces user-specific access levels for users who authenticate for management access (see the aaa authentication console LOCAL command). Without this command, the ASASM only supports privilege levels for local database users and defaults all other types of users to level 15. This command also enables management authorization for local, RADIUS, LDAP (mapped), and TACACS+ users. Use the aaa authorization exec LOCAL command to enable attributes to be taken from the local database. See the “Limiting User CLI and ASDM Access with Management Authorization” section on page 34-20 for information about configuring a user on a AAA server to accommodate management authorization.</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# aaa authorization exec authentication-server</td>
<td></td>
</tr>
</tbody>
</table>
Examples

The `filter` command has the following forms:

- `filter` (represented by the `configure` option)
- `show running-config filter`
- `clear configure filter`

You can set the privilege level separately for each form, or set the same privilege level for all forms by omitting this option. The following example shows how to set each form separately:

```
hostname(config)# privilege show level 5 command filter
hostname(config)# privilege clear level 10 command filter
hostname(config)# privilege cmd level 10 command filter
```

Alternatively, the following example shows how to set all filter commands to the same level:

```
hostname(config)# privilege level 5 command filter
```

The `show privilege` command separates the forms in the display.

The following example shows the use of the `mode` keyword. The `enable` command must be entered from user EXEC mode, while the `enable password` command, which is accessible in configuration mode, requires the highest privilege level:

```
hostname(config)# privilege cmd level 0 mode enable command enable
hostname(config)# privilege cmd level 15 mode cmd command enable
hostname(config)# privilege show level 15 mode cmd command enable
```

The following example shows an additional command, the `configure` command, which uses the `mode` keyword:

```
hostname(config)# privilege show level 5 mode cmd command configure
hostname(config)# privilege clear level 15 mode cmd command configure
hostname(config)# privilege cmd level 15 mode cmd command configure
hostname(config)# privilege cmd level 15 mode enable command configure
```

**Note**

This last line is for the `configure terminal` command.
Viewing Local Command Privilege Levels

The following commands let you view privilege levels for commands.

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show running-config all privilege all</td>
<td>Shows all commands.</td>
</tr>
<tr>
<td>show running-config privilege level level</td>
<td>Shows commands for a specific level. The level is an integer between 0 and 15.</td>
</tr>
<tr>
<td>show running-config privilege command command</td>
<td>Shows the level of a specific command.</td>
</tr>
</tbody>
</table>

Examples

For the show running-config all privilege all command, the ASASM displays the current assignment of each CLI command to a privilege level. The following is sample output from this command:

```
hostname(config)# show running-config all privilege all
privilege show level 15 command aaa
privilege clear level 15 command aaa
privilege configure level 15 command aaa
privilege show level 15 command aaa-server
privilege clear level 15 command aaa-server
privilege configure level 15 command aaa-server
privilege show level 15 command access-group
privilege clear level 15 command access-group
privilege configure level 15 command access-group
privilege show level 15 command access-list
privilege clear level 15 command access-list
privilege configure level 15 command access-list
privilege show level 15 command activation-key
privilege configure level 15 command activation-key
....
```

The following example displays the command assignments for privilege level 10:

```
hostname(config)# show running-config privilege level 10
privilege show level 15 command aaa
```

The following example displays the command assignments for the access-list command:

```
hostname(config)# show running-config privilege command access-list
privilege show level 15 command access-list
privilege clear level 15 command access-list
privilege configure level 15 command access-list
```

Configuring Commands on the TACACS+ Server

You can configure commands on a Cisco Secure Access Control Server (ACS) TACACS+ server as a shared profile component, for a group, or for individual users. For third-party TACACS+ servers, see your server documentation for more information about command authorization support.

See the following guidelines for configuring commands in Cisco Secure ACS Version 3.1; many of these guidelines also apply to third-party servers:

- The ASASM sends the commands to be authorized as shell commands, so configure the commands on the TACACS+ server as shell commands.
Note

Cisco Secure ACS might include a command type called “pix-shell.” Do not use this type for ASASM command authorization.

- The first word of the command is considered to be the main command. All additional words are considered to be arguments, which need to be preceded by permit or deny.

  For example, to allow the `show running-configuration aaa-server` command, add `show running-configuration` to the command field, and type `permit aaa-server` in the arguments field.

- You can permit all arguments of a command that you do not explicitly deny by checking the Permit Unmatched Args check box.

  For example, you can configure just the `show` command, and then all the `show` commands are allowed. We recommend using this method so that you do not have to anticipate every variant of a command, including abbreviations and ?, which shows CLI usage (see Figure 34-1).

Figure 34-1  Permitting All Related Commands

- For commands that are a single word, you must permit unmatched arguments, even if there are no arguments for the command, for example enable or help (see Figure 34-2).

Figure 34-2  Permitting Single Word Commands

- To disallow some arguments, enter the arguments preceded by deny.
For example, to allow enable, but not enable password, enter enable in the commands field, and deny password in the arguments field. Be sure to check the Permit Unmatched Args check box so that enable alone is still allowed (see Figure 34-3).

**Figure 34-3 Disallowing Arguments**

- When you abbreviate a command at the command line, the ASASM expands the prefix and main command to the full text, but it sends additional arguments to the TACACS+ server as you enter them.

  For example, if you enter sh log, then the ASASM sends the entire command to the TACACS+ server, show logging. However, if you enter sh log mess, then the ASASM sends show logging mess to the TACACS+ server, and not the expanded command show logging message. You can configure multiple spellings of the same argument to anticipate abbreviations (see Figure 34-4).

**Figure 34-4 Specifying Abbreviations**

- We recommend that you allow the following basic commands for all users:
  - show checksum
  - show curpriv
  - enable
  - help
  - show history
Configuring AAA for System Administrators

- login
- logout
- pager
- show pager
- clear pager
- quit
- show version

Configuring TACACS+ Command Authorization

If you enable TACACS+ command authorization, and a user enters a command at the CLI, the ASASM sends the command and username to the TACACS+ server to determine if the command is authorized.

Before you enable TACACS+ command authorization, be sure that you are logged into the ASASM as a user that is defined on the TACACS+ server, and that you have the necessary command authorization to continue configuring the ASASM. For example, you should log in as an admin user with all commands authorized. Otherwise, you could become unintentionally locked out.

Do not save your configuration until you are sure that it works the way you want. If you get locked out because of a mistake, you can usually recover access by restarting the ASASM. If you still get locked out, see the “Recovering from a Lockout” section on page 34-30.

Be sure that your TACACS+ system is completely stable and reliable. The necessary level of reliability typically requires that you have a fully redundant TACACS+ server system and fully redundant connectivity to the ASASM. For example, in your TACACS+ server pool, include one server connected to interface 1, and another to interface 2. You can also configure local command authorization as a fallback method if the TACACS+ server is unavailable. In this case, you need to configure local users and command privilege levels according to procedures listed in the “Configuring Command Authorization” section on page 34-21.

To configure TACACS+ command authorization, enter the following command:

**Detailed Steps**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa authorization command</td>
<td>Performs command authorization using a TACACS+ server.</td>
</tr>
<tr>
<td>tacacs+_server_group [LOCAL]</td>
<td>You can configure the ASASM to use the local database as a fallback method if the TACACS+ server is unavailable. To enable fallback, specify the server group name followed by LOCAL (LOCAL is case sensitive).</td>
</tr>
<tr>
<td>Example:</td>
<td>We recommend that you use the same username and password in the local database as the TACACS+ server because the ASASM prompt does not give any indication which method is being used. Be sure to configure users in the local database (see the “Adding a User Account to the Local Database” section on page 33-19) and command privilege levels (see the “Configuring Local Command Authorization” section on page 34-22).</td>
</tr>
</tbody>
</table>
Configuring Management Access Accounting

You can send accounting messages to the TACACS+ accounting server when you enter any command other than show commands at the CLI. You can configure accounting when users log in, when they enter the enable command, or when they issue commands.

For command accounting, you can only use TACACS+ servers.

To configure management access and enable command accounting, perform the following steps:

**Detailed Steps**

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>`aaa accounting {serial</td>
<td>telnet</td>
</tr>
<tr>
<td>Example:</td>
<td><code>hostname(config)# aaa accounting telnet console group_1</code></td>
<td></td>
</tr>
<tr>
<td>Step 2</td>
<td><code>aaa accounting command [privilege level] server-tag</code></td>
<td>Enables command accounting. Only TACACS+ servers support command accounting. Where privilege level is the minimum privilege level and server-tag is the name of the TACACS+ server group to which the ASASM should send command accounting messages.</td>
</tr>
<tr>
<td>Example:</td>
<td><code>hostname(config)# aaa accounting command privilege 15 group_1</code></td>
<td></td>
</tr>
</tbody>
</table>

**Viewing the Currently Logged-In User**

To view the current logged-in user, enter the following command:

`hostname# show curpriv`

The following is sample output from the show curpriv command:

```
hostname# show curpriv
Username: admin
Current privilege level: 15
Current Mode/s: P_PRIV
```

**Table 34-1** describes the show curpriv command output.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Username</td>
<td>Username. If you are logged in as the default user, the name is enable_1 (user EXEC) or enable_15 (privileged EXEC).</td>
</tr>
</tbody>
</table>
Recovering from a Lockout

In some circumstances, when you turn on command authorization or CLI authentication, you can be locked out of the ASASM CLI. You can usually recover access by restarting the ASASM. However, if you already saved your configuration, you might be locked out. Table 34-2 lists the common lockout conditions and how you might recover from them.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Lockout Condition</th>
<th>Description</th>
<th>Workaround: Single Mode</th>
<th>Workaround: Multiple Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local CLI authentication</td>
<td>No users in the local database</td>
<td>If you have no users in the local database, you cannot log in, and you cannot add any users.</td>
<td>Log in and reset the passwords and aaa commands.</td>
<td>Session into the ASASM from the switch. From the system execution space, you can change to the context and add a user.</td>
</tr>
<tr>
<td>TACACS+ command authorization</td>
<td>Server down or unreachable and you do not have the fallback method configured</td>
<td>If the server is unreachable, then you cannot log in or enter any commands.</td>
<td>1. Log in and reset the passwords and AAA commands.</td>
<td>1. If the server is unreachable because the network configuration is incorrect on the ASASM, session into the ASASM from the switch. From the system execution space, you can change to the context and reconfigure your network settings.</td>
</tr>
<tr>
<td>TACACS+ CLI authentication</td>
<td></td>
<td></td>
<td>2. Configure the local database as a fallback method so you do not get locked out when the server is down.</td>
<td>2. Configure the local database as a fallback method so you do not get locked out when the server is down.</td>
</tr>
<tr>
<td>RADIUS CLI authentication</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Configuring AAA for System Administrators

Setting a Management Session Quota

An administrator can establish a maximum number of simultaneous management sessions. If the maximum is reached, no additional sessions are allowed and a syslog message is generated. To prevent a system lockout, the management session quota mechanism cannot block a console session.

To set a management session maximum, enter the following command:

```
quota management-session number
```

**Example:**

```
hostname(config)# quota management-session 1000
```

Sets the maximum number of simultaneous ASDM, SSH, and Telnet sessions that are allowed on the ASASM. The `no` form of this command sets the quota value to 0, which means that there is no session limit.
## Feature History for Management Access

Table 34-3 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Management Access</td>
<td>7.0(1)</td>
<td>We introduced this feature. We introduced the following commands: show running-config all privilege all, show running-config privilege level, show running-config privilege command, telnet, telnet timeout, ssh, ssh timeout, http, http server enable, asdm image disk, banner, console timeout, icmp, ipv6 icmp, management access, aaa authentication console, aaa authentication enable console, aaa authentication telnet</td>
</tr>
</tbody>
</table>
Configuring AAA Rules for Network Access

This chapter describes how to enable AAA (pronounced “triple A”) for network access. For information about AAA for management access, see the “Configuring AAA for System Administrators” section on page 34-12.

This chapter includes the following sections:

- AAA Performance, page 35-1
- Licensing Requirements for AAA Rules, page 35-1
- Guidelines and Limitations, page 35-2
- Configuring Authentication for Network Access, page 35-2
- Configuring Authorization for Network Access, page 35-11
- Configuring Accounting for Network Access, page 35-18
- Using MAC Addresses to Exempt Traffic from Authentication and Authorization, page 35-20
- Feature History for AAA Rules, page 35-21

AAA Performance

The ASASM uses “cut-through proxy” to significantly improve performance compared to a traditional proxy server. The performance of a traditional proxy server suffers because it analyzes every packet at the application layer of the OSI model. The ASASM cut-through proxy challenges a user initially at the application layer and then authenticates with standard AAA servers or the local database. After the ASASM authenticates the user, it shifts the session flow, and all traffic flows directly and quickly between the source and destination while maintaining session state information.

Licensing Requirements for AAA Rules

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>
Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

**Context Mode Guidelines**
Supported in single and multiple context mode.

**Firewall Mode Guidelines**
Supported in routed and transparent firewall mode.

**IPv6 Guidelines**
Supports IPv6.

Configuring Authentication for Network Access

This section includes the following topics:

- Information About Authentication, page 35-2
- Configuring Network Access Authentication, page 35-4
- Enabling Secure Authentication of Web Clients, page 35-6
- Authenticating Directly with the ASASM, page 35-7

Information About Authentication

The ASASM lets you configure network access authentication using AAA servers. This section includes the following topics:

- One-Time Authentication, page 35-2
- Applications Required to Receive an Authentication Challenge, page 35-2
- ASASM Authentication Prompts, page 35-3
- Static PAT and HTTP, page 35-4

One-Time Authentication

A user at a given IP address only needs to authenticate one time for all rules and types, until the authentication session expires. (See the `timeout uauth` command in the command reference for timeout values.) For example, if you configure the ASASM to authenticate Telnet and FTP, and a user first successfully authenticates for Telnet, then as long as the authentication session exists, the user does not also have to authenticate for FTP.

Applications Required to Receive an Authentication Challenge

Although you can configure the ASASM to require authentication for network access to any protocol or service, users can authenticate directly with HTTP, HTTPS, Telnet, or FTP only. A user must first authenticate with one of these services before the ASASM allows other traffic requiring authentication.

The authentication ports that the ASASM supports for AAA are fixed as follows:
- Port 21 for FTP
- Port 23 for Telnet
- Port 80 for HTTP
- Port 443 for HTTPS

**ASASM Authentication Prompts**

For Telnet and FTP, the ASASM generates an authentication prompt.

For HTTP, the ASASM uses basic HTTP authentication by default, and provides an authentication prompt. You can optionally configure the ASASM to redirect users to an internal web page where they can enter their username and password (configured with the `aaa authentication listener` command).

For HTTPS, the ASASM generates a custom login screen. You can optionally configure the ASASM to redirect users to an internal web page where they can enter their username and password (configured with the `aaa authentication listener` command).

Redirection is an improvement over the basic method because it provides an improved user experience when authenticating, and an identical user experience for HTTP and HTTPS in both Easy VPN and firewall modes. It also supports authenticating directly with the ASASM.

You might want to continue to use basic HTTP authentication for the following reasons:

- You do not want the ASASM to open listening ports.
- You use NAT on a router and you do not want to create a translation rule for the web page served by the ASASM.
- Basic HTTP authentication might work better with your network.

For example non-browser applications, as when a URL is embedded in e-mail, might be more compatible with basic authentication.

After you authenticate correctly, the ASASM redirects you to your original destination. If the destination server also has its own authentication, the user enters another username and password. If you use basic HTTP authentication and need to enter another username and password for the destination server, then you need to configure the `virtual http` command.

**Note**

If you use HTTP authentication, by default the username and password are sent from the client to the ASASM in clear text; in addition, the username and password are sent on to the destination web server as well. See the “Enabling Secure Authentication of Web Clients” section on page 35-6 for information to secure your credentials.

For FTP, a user has the option of entering the ASASM username followed by an at sign (@) and then the FTP username (name1@name2). For the password, the user enters the ASASM password followed by an at sign (@) and then the FTP password (password1@password2). For example, enter the following text:

```
name> name1@name2
password> password1@password2
```

This feature is useful when you have cascaded firewalls that require multiple logins. You can separate several names and passwords by multiple at signs (@).
### Static PAT and HTTP

For HTTP authentication, the ASASM checks real ports when static PAT is configured. If it detects traffic destined for real port 80, regardless of the mapped port, the ASASM intercepts the HTTP connection and enforces authentication.

For example, assume that outside TCP port 889 is translated to port 80 and that any relevant access lists permit the traffic:

```plaintext
object network obj-192.168.123.10-01
    host 192.168.123.10
    nat (inside,outside) static 10.48.66.155 service tcp 80 889
```

Then when users try to access 10.48.66.155 on port 889, the ASASM intercepts the traffic and enforces HTTP authentication. Users see the HTTP authentication page in their web browsers before the ASASM allows HTTP connection to complete.

If the local port is different than port 80, as in the following example:

```plaintext
object network obj-192.168.123.10-02
    host 192.168.123.10
    nat (inside,outside) static 10.48.66.155 service tcp 111 889
```

Then users do not see the authentication page. Instead, the ASASM sends an error message to the web browser indicating that the user must be authenticated before using the requested service.

### Configuring Network Access Authentication

To configure network access authentication, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><strong>aaa-server</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><em>hostname(config)# aaa-server AuthOutbound protocol tacacs</em></td>
</tr>
<tr>
<td></td>
<td>Identifies your AAA servers. If you have already identified them, continue to the next step. For more information about identifying AAA servers, see the “Configuring AAA Server Groups” section on page 33-11.</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><strong>access-list</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><em>hostname(config)# access-list MAIL_AUTH extended permit tcp any any eq smtp</em></td>
</tr>
<tr>
<td></td>
<td>Creates an access list that identifies the source addresses and destination addresses of traffic you want to authenticate. For details, see Chapter 14, “Adding an Extended Access List.”</td>
</tr>
<tr>
<td></td>
<td>The <strong>permit</strong> ACEs mark matching traffic for authentication, while <strong>deny</strong> entries exclude matching traffic from authentication. Be sure to include the destination ports for either HTTP, HTTPS, Telnet, or FTP in the access list, because the user must authenticate with one of these services before other services are allowed through the ASASM.</td>
</tr>
</tbody>
</table>
### Command Purpose

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 3  <code>aaa authentication match acl_name interface_name</code> <code>server_group</code></td>
<td>Configures authentication. The <code>acl_name</code> argument is the name of the access list that you created in Step 2. The <code>interface_name</code> argument is the name of the interface specified with the <code>nameif</code> command. The <code>server_group</code> argument is the AAA server group that you created in Step 1.</td>
</tr>
<tr>
<td>Example: <code>hostname(config)# aaa authentication match MAIL_AUTH inside AuthOutbound</code></td>
<td></td>
</tr>
<tr>
<td>Step 4  <code>aaa authentication listener http[s] interface_name</code> <code>[port portnum] redirect</code></td>
<td>(Optional) Enables the redirection method of authentication for HTTP or HTTPS connections. The <code>interface_name</code> argument is the interface on which you want to enable listening ports. The <code>port</code> <code>portnum</code> argument specifies the port number on which the ASASM listens; the defaults are 80 (HTTP) and 443 (HTTPS). You can use any port number and retain the same functionality, but be sure your direct authentication users know the port number; redirected traffic is sent to the correct port number automatically, but direct authenticators must specify the port number manually. Enter this command separately for HTTP and for HTTPS.</td>
</tr>
<tr>
<td>Example: <code>hostname(config)# aaa authentication listener http inside redirect</code></td>
<td></td>
</tr>
<tr>
<td>Step 5  <code>aaa local authentication attempts max-fail number</code></td>
<td>(Optional) Uses the local database for network access authentication and limits the number of consecutive failed login attempts that the ASASM allows any given user account (with the exception of users with a privilege level of 15. This feature does not affect level 15 users). The <code>number</code> argument value is between 1 and 16.</td>
</tr>
<tr>
<td>Example: <code>hostname(config)# aaa local authentication attempts max-fail 7</code></td>
<td>Tip To clear the lockout status of a specific user or all users, use the <code>clear aaa local user lockout</code> command.</td>
</tr>
</tbody>
</table>

### Examples

The following example authenticates all inside HTTP traffic and SMTP traffic:

```
hostname(config)# aaa-server AuthOutbound protocol tacacs+
hostname(config-aaa-server-group)# exit
hostname(config)# aaa-server AuthOutbound (inside) host 10.1.1.1
hostname(config-aaa-server-host)# key TACPlusUauthKey
hostname(config-aaa-server-host)# exit
hostname(config)# access-list MAIL_AUTH extended permit tcp any eq smtp
hostname(config)# access-list MAIL_AUTH extended permit tcp any eq www
hostname(config)# aaa authentication match MAIL_AUTH inside AuthOutbound
```
hostname(config)# aaa authentication listener http inside redirect

The following example authenticates Telnet traffic from the outside interface to a particular server (209.165.201.5):

hostname(config)# aaa-server AuthInbound protocol tacacs+
hostname(config-aaa-server-group)# exit
hostname(config)# aaa-server AuthInbound (inside) host 10.1.1.1
hostname(config-aaa-server-host)# key TACPlusUauthKey
hostname(config-aaa-server-host)# exit
hostname(config)# access-list TELNET_AUTH extended permit tcp any host 209.165.201.5 eq telnet
hostname(config)# aaa authentication match TELNET_AUTH outside AuthInbound

For more information about authentication, see the “Information About Authentication” section on page 35-2.

Enabling Secure Authentication of Web Clients

If you use HTTP authentication, by default the username and password are sent from the client to the ASASM in clear text; in addition, the username and password are sent to the destination web server as well.

The ASASM provides the following methods for securing HTTP authentication:

- Enable the redirection method of authentication for HTTP—Use the `aaa authentication listener` command with the `redirect` keyword. This method prevents the authentication credentials from continuing to the destination server. See the “ASASM Authentication Prompts” section on page 35-3 for more information about the redirection method compared to the basic method.

- Enable virtual HTTP—Use the `virtual http` command to authenticate separately with the ASASM and with the HTTP server. Even if the HTTP server does not need a second authentication, this command achieves the effect of stripping the basic authentication credentials from the HTTP GET request. See the “Authenticating HTTP(S) Connections with a Virtual Server” section on page 35-8 for more information.

Enable the exchange of usernames and passwords between a web client and the ASASM with HTTPS—Use the `aaa authentication secure-http-client` command to enable the exchange of usernames and passwords between a web client and the ASASM with HTTPS. This is the only method that protects credentials between the client and the ASASM, as well as between the ASASM and the destination server. You can use this method alone, or in conjunction with either of the other methods so you can maximize your security.

After enabling this feature, when a user requires authentication when using HTTP, the ASASM redirects the HTTP user to an HTTPS prompt. After you authenticate correctly, the ASASM redirects you to the original HTTP URL.

Secured, web-client authentication has the following limitations:

- A maximum of 16 concurrent HTTPS authentication sessions are allowed. If all 16 HTTPS authentication processes are running, a new connection requiring authentication will not succeed.

- When `uauth timeout 0` is configured (the `uauth timeout` is set to 0), HTTPS authentication might not work. If a browser initiates multiple TCP connections to load a web page after HTTPS authentication, the first connection is let through, but the subsequent connections trigger authentication. As a result, users are continuously presented with an authentication page, even if the correct username and password are entered each time. To work around this, set the `uauth`
timeout to 1 second with the timeout uauth 0:0:1 command. However, this workaround opens a 1-second window of opportunity that might allow unauthenticated users to go through the firewall if they are coming from the same source IP address.

Because HTTPS authentication occurs on the SSL port 443, users must not configure an access-list command statement to block traffic from the HTTP client to the HTTP server on port 443. Furthermore, if static PAT is configured for web traffic on port 80, it must also be configured for the SSL port.

- In the following example, the first set of commands configures static PAT for web traffic, and the second set of commands must be added to support the HTTPS authentication configuration:

```plaintext
object network obj-10.130.16.10-01
  host 10.130.16.10
  nat (inside,outside) static 10.132.16.200 service tcp 80 80
object network obj-10.130.16.10-02
  host 10.130.16.10
  nat (inside,outside) static 10.132.16.200 service tcp 443 443
```

**Authenticating Directly with the ASASM**

If you do not want to allow HTTP, HTTPS, Telnet, or FTP through the ASASM but want to authenticate other types of traffic, you can authenticate with the ASASM directly using HTTP, HTTPS, or Telnet.

This section includes the following topics:

- Authenticating HTTP(S) Connections with a Virtual Server, page 35-8
- Authenticating Telnet Connections with a Virtual Server, page 35-9
### Authenticating HTTP(S) Connections with a Virtual Server

If you enabled the redirection method of HTTP and HTTPS authentication in the “Configuring Network Access Authentication” section on page 35-4, then you have also automatically enabled direct authentication.

When you use HTTP authentication on the ASASM (see the “Configuring Network Access Authentication” section on page 35-4), the ASASM uses basic HTTP authentication by default.

To continue to use basic HTTP authentication, and to enable direct authentication for HTTP and HTTPS, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>aaa authentication listener http[s] [port portnum] redirect</strong></td>
<td>(Optional) Enables the redirection method of authentication for HTTP or HTTPS connections. The <em>interface_name</em> argument is the interface on which you want to enable listening ports. The <em>port portnum</em> argument specifies the port number on which the ASASM listens; the defaults are 80 (HTTP) and 443 (HTTPS). You can use any port number and retain the same functionality, but be sure your direct authentication users know the port number; redirected traffic is sent to the correct port number automatically, but direct authenticators must specify the port number manually. Enter this command separately for HTTP and for HTTPS.</td>
</tr>
</tbody>
</table>

Example:

```
hostname(config)# aaa authentication listener http inside redirect
```

If the destination HTTP server requires authentication in addition to the ASASM, then to authenticate separately with the ASASM (via a AAA server) and with the HTTP server, enter the following command:
Configuring Authentication for Network Access

Chapter 35      Configuring AAA Rules for Network Access

Cisco ASA Services Module CLI Configuration Guide

Authenticating Telnet Connections with a Virtual Server

Although you can configure network access authentication for any protocol or service (see the `aaa` authentication match or `aaa authentication include` command), you can authenticate directly with HTTP, Telnet, or FTP only. A user must first authenticate with one of these services before other traffic that requires authentication is allowed through. If you do not want to allow HTTP, Telnet, or FTP traffic through the ASASM, but want to authenticate other types of traffic, you can configure virtual Telnet; the user Telnets to a given IP address configured on the ASASM, and the ASASM issues a Telnet prompt.

To configure a virtual Telnet server, enter the following command:

```
hostname(config)# virtual http
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| virtual http     | Redirects all HTTP connections that require AAA authentication to the virtual HTTP server on the ASASM. The ASASM prompts for the AAA server username and password. After the AAA server authenticates the user, the ASASM redirects the HTTP connection back to the original server, but it does not include the AAA server username and password. Because the username and password are not included in the HTTP packet, the HTTP server prompts the user separately for the HTTP server username and password. For inbound users (from lower security to higher security), you must also include the virtual HTTP address as a destination interface in the access list applied to the source interface. In addition, you must add a static NAT command for the virtual HTTP IP address, even if NAT is not required. An identity NAT command is typically used (where you translate the address to itself). For outbound users, there is an explicit permit for traffic, but if you apply an access list to an inside interface, be sure to allow access to the virtual HTTP address. A static statement is not required. Do not set the `timeout uauth` command duration to 0 seconds when using the `virtual http` command, because this setting prevents HTTP connections to the actual web server. You can authenticate directly with the ASASM at the following URLs when you enable AAA for the interface:

```
http://interface_ip[port]/netaccess/connstatus.html
https://interface_ip[port]/netaccess/connstatus.html
```

Without virtual HTTP, the same username and password that you used to authenticate with the ASASM are sent to the HTTP server; you are not prompted separately for the HTTP server username and password. Assuming the username and password are not the same for the AAA and HTTP servers, then the HTTP authentication fails.

You can authenticate directly with the ASASM at the following URLs when you enable AAA for the interface:
Chapter 35      Configuring AAA Rules for Network Access

Configuring Authentication for Network Access

Examples

The following example shows how to enable virtual Telnet together with AAA authentication for other services:

```
hostname(config)# virtual telnet 209.165.202.129
hostname(config)# access-list ACL-IN extended permit tcp any host 209.165.200.225 eq smtp
hostname(config)# access-list ACL-IN remark This is the SMTP server on the inside
hostname(config)# access-list ACL-IN extended permit tcp any host 209.165.202.129 eq telnet
hostname(config)# access-group ACL-IN in interface outside
hostname(config)# nat (inside,outside) static 209.165.202.129
hostname(config)# access-list AUTH extended permit tcp any host 209.165.200.225 eq smtp
hostname(config)# access-list AUTH remark This is the SMTP server on the inside
hostname(config)# access-list AUTH extended permit tcp any host 209.165.202.129 eq telnet
hostname(config)# access-list AUTH remark This is the virtual Telnet address
hostname(config)# aaa authentication match AUTH outside tacacs+
```
Configuring Authorization for Network Access

After a user authenticates for a given connection, the ASASM can use authorization to further control traffic from the user.

This section includes the following topics:

- Configuring TACACS+ Authorization, page 35-11
- Configuring RADIUS Authorization, page 35-14

Configuring TACACS+ Authorization

You can configure the ASASM to perform network access authorization with TACACS+. You identify the traffic to be authorized by specifying access lists that authorization rules must match. Alternatively, you can identify the traffic directly in authorization rules themselves.

Tip

Using access lists to identify traffic to be authorized can greatly reduced the number of authorization commands that you must enter. This is because each authorization rule that you enter can specify only one source and destination subnet and service, whereas an access list can include many entries.

Authentication and authorization statements are independent; however, any unauthenticated traffic matched by an authorization rule will be denied. For authorization to succeed:

1. A user must first authenticate with the ASASM.
   
   Because a user at a given IP address only needs to authenticate one time for all rules and types, if the authentication session has not expired, authorization can occur even if the traffic is not matched by an authentication rule.

2. After a user authenticates, the ASASM checks the authorization rules for matching traffic.

3. If the traffic matches the authorization rule, the ASASM sends the username to the TACACS+ server.

4. The TACACS+ server responds to the ASASM with a permit or a deny for that traffic, based on the user profile.

5. The ASASM enforces the authorization rule in the response.

See the documentation for your TACACS+ server for information about configuring network access authorizations for a user.

To configure TACACS+ authorization, perform the following steps:
### Chapter 35  Configuring AAA Rules for Network Access

#### Configuring Authorization for Network Access

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Identifies your AAA servers. If you have already identified them, continue to the next step. For more information about identifying AAA servers, see the “Configuring AAA Server Groups” section on page 33-11.</td>
</tr>
<tr>
<td><code>aaa-server</code></td>
<td>Example:</td>
</tr>
<tr>
<td>hostname(config)# aaa-server AuthOutbound protocol tacacs+</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Creates an access list that identifies the source addresses and destination addresses of traffic you want to authenticate. For details, see Chapter 14, “Adding an Extended Access List.” The <code>permit</code> ACEs mark matching traffic for authentication, while <code>deny</code> entries exclude matching traffic from authentication. Be sure to include the destination ports for either HTTP, HTTPS, Telnet, or FTP in the access list, because the user must authenticate with one of these services before other services are allowed through the ASASM.</td>
</tr>
<tr>
<td><code>access-list</code></td>
<td>Example:</td>
</tr>
<tr>
<td>hostname(config)# access-list MAIL_AUTH extended permit tcp any any eq smtp</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>Configures authentication. The <code>acl_name</code> argument is the name of the access list that you created in Step 2., The <code>interface_name</code> argument is the name of the interface specified with the <code>nameif</code> command, and the <code>server_group</code> argument is the AAA server group that you created in Step 1.</td>
</tr>
<tr>
<td><code>aaa authentication match acl_name interface_name server_group</code></td>
<td>Example:</td>
</tr>
<tr>
<td>hostname(config)# aaa authentication match MAIL_AUTH inside AuthOutbound</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>(Optional) Enables the redirection method of authentication for HTTP or HTTPS connections. The <code>interface_name</code> argument is the interface on which you want to enable listening ports. The <code>port</code> argument specifies the port number on which the ASASM listens; the defaults are 80 (HTTP) and 443 (HTTPS). You can use any port number and retain the same functionality, but be sure your direct authentication users know the port number; redirected traffic is sent to the correct port number automatically, but direct authenticators must specify the port number manually. Enter this command separately for HTTP and for HTTPS.</td>
</tr>
<tr>
<td><code>aaa authentication listener http[s] interface_name [port portnum] redirect</code></td>
<td>Example:</td>
</tr>
<tr>
<td>hostname(config)# aaa authentication listener http inside redirect</td>
<td></td>
</tr>
</tbody>
</table>
### Step 5

**Command:**

```bash
aaa local authentication attempts max-fail number
```

**Purpose:**

(Optional) Uses the local database for network access authentication and limits the number of consecutive failed login attempts that the ASASM allows any given user account (with the exception of users with a privilege level of 15). This feature does not affect level 15 users. The `number` argument value is between 1 and 16.

**Example:**

```
hostname(config)# aaa local authentication attempts max-fail 7
```

**Tip**

To clear the lockout status of a specific user or all users, use the `clear aaa local user lockout` command.

### Step 6

**Command:**

```bash
access-list
```

**Purpose:**

Create an access list that identifies the source addresses and destination addresses of traffic that you want to authorize. For instructions, see Chapter 14, “Adding an Extended Access List.”

**Example:**

```
hostname(config)# access-list TELNET_AUTH extended permit tcp any any eq telnet
```

### Step 7

**Command:**

```bash
aaa authorization match acl_name interface_name server_group
```

**Purpose:**

Enables authorization.

**Example:**

```
hostname(config)# aaa authorization match TELNET_AUTH inside AuthOutbound
```

**Note**

Alternatively, you can use the `aaa authorization include` command (which identifies traffic within the command) but you cannot use both methods in the same configuration. See the command reference for more information.

### Examples

The following example authenticates and authorizes inside Telnet traffic. Telnet traffic to servers other than 209.165.201.5 can be authenticated alone, but traffic to 209.165.201.5 requires authorization.

```
hostname(config)# access-list TELNET_AUTH extended permit tcp any any eq telnet
hostname(config)# access-list SERVER_AUTH extended permit tcp any any eq telnet
hostname(config)# aaa-server AuthOutbound protocol tacacs+
hostname(config-aaa-server-group)# exit
hostname(config)# aaa-server AuthOutbound (inside) host 10.1.1.1
```
hostname(config-aaa-server-host)# key TACPlusUauthKey
hostname(config-aaa-server-host)# exit
hostname(config)# aaa authentication match TELNET_AUTH inside AuthOutbound
hostname(config)# aaa authorization match SERVER_AUTH inside AuthOutbound

### Configuring RADIUS Authorization

When authentication succeeds, the RADIUS protocol returns user authorizations in the access-accept message sent by a RADIUS server. For more information about configuring authentication, see the “Configuring Network Access Authentication” section on page 35-4.

When you configure the ASASM to authenticate users for network access, you are also implicitly enabling RADIUS authorizations; therefore, this section contains no information about configuring RADIUS authorization on the ASASM. It does provide information about how the ASASM handles access list information received from RADIUS servers.

You can configure a RADIUS server to download an access list to the ASASM or an access list name at the time of authentication. The user is authorized to do only what is permitted in the user-specific access list.

**Note**
If you have used the `access-group` command to apply access lists to interfaces, be aware of the following effects of the `per-user-override` keyword on authorization by user-specific access lists:

- Without the `per-user-override` keyword, traffic for a user session must be permitted by both the interface access list and the user-specific access list.
- With the `per-user-override` keyword, the user-specific access list determines what is permitted.

For more information, see the `access-group` command entry in the command reference.

This section includes the following topics:

- Configuring a RADIUS Server to Send Downloadable Access Control Lists, page 35-14
- Configuring a RADIUS Server to Download Per-User Access Control List Names, page 35-18

### Configuring a RADIUS Server to Send Downloadable Access Control Lists

This section describes how to configure Cisco Secure ACS or a third-party RADIUS server and includes the following topics:

- About the Downloadable Access List Feature and Cisco Secure ACS, page 35-14
- Configuring Cisco Secure ACS for Downloadable Access Lists, page 35-16
- Configuring Any RADIUS Server for Downloadable Access Lists, page 35-17
- Converting Wildcard Netmask Expressions in Downloadable Access Lists, page 35-18

### About the Downloadable Access List Feature and Cisco Secure ACS

Downloadable access lists is the most scalable means of using Cisco Secure ACS to provide the appropriate access lists for each user. It provides the following capabilities:

- Unlimited access list size—Downloadable access lists are sent using as many RADIUS packets as required to transport the full access list from Cisco Secure ACS to the ASASM.
Simplified and centralized management of access lists—Downloadable access lists enable you to write a set of access lists once and apply it to many user or group profiles and distribute it to many ASASMs.

This approach is most useful when you have very large access list sets that you want to apply to more than one Cisco Secure ACS user or group; however, its ability to simplify Cisco Secure ACS user and group management makes it useful for access lists of any size.

The ASASM receives downloadable access lists from Cisco Secure ACS using the following process:

1. The ASASM sends a RADIUS authentication request packet for the user session.
2. If Cisco Secure ACS successfully authenticates the user, Cisco Secure ACS returns a RADIUS access-accept message that includes the internal name of the applicable downloadable access list. The Cisco IOS cisco-av-pair RADIUS VSA (vendor 9, attribute 1) includes the following attribute-value pair to identify the downloadable access list set:
   
   ACS:CiscoSecure-Defined-ACL=acl-set-name

   where acl-set-name is the internal name of the downloadable access list, which is a combination of the name assigned to the access list by the Cisco Secure ACS administrator and the date and time that the access list was last modified.
3. The ASASM examines the name of the downloadable access list and determines if it has previously received the named downloadable access list.
   - If the ASASM has previously received the named downloadable access list, communication with Cisco Secure ACS is complete and the ASASM applies the access list to the user session. Because the name of the downloadable access list includes the date and time that it was last modified, matching the name sent by Cisco Secure ACS to the name of an access list previously downloaded means that the ASASM has the most recent version of the downloadable access list.
   - If the ASASM has not previously received the named downloadable access list, it may have an out-of-date version of the access list or it may not have downloaded any version of the access list. In either case, the ASASM issues a RADIUS authentication request using the downloadable access list name as the username in the RADIUS request and a null password attribute. In a cisco-av-pair RADIUS VSA, the request also includes the following attribute-value pairs:

   AAA:service=ip-admission
   AAA:event=acl-download

   In addition, the ASASM signs the request with the Message-Authenticator attribute (IETF RADIUS attribute 80).

4. After receipt of a RADIUS authentication request that has a username attribute that includes the name of a downloadable access list, Cisco Secure ACS authenticates the request by checking the Message-Authenticator attribute. If the Message-Authenticator attribute is missing or incorrect, Cisco Secure ACS ignores the request. The presence of the Message-Authenticator attribute prevents malicious use of a downloadable access list name to gain unauthorized network access. The Message-Authenticator attribute and its use are defined in RFC 2869, RADIUS Extensions, available at http://www.ietf.org.

5. If the access list required is less than approximately 4 KB in length, Cisco Secure ACS responds with an access-accept message that includes the access list. The largest access list that can fit in a single access-accept message is slightly less than 4 KB, because part of the message must be other required attributes.

   Cisco Secure ACS sends the downloadable access list in a cisco-av-pair RADIUS VSA. The access list is formatted as a series of attribute-value pairs that each include an ACE and are numbered serially:

   ip:inacl#1=ACE-1
The following example is of an attribute-value pair:

```
ip:inacl#1=permit tcp 10.1.0.0 255.0.0.0 10.0.0.0 255.0.0.0
```

6. If the access list required is more than approximately 4 KB in length, Cisco Secure ACS responds with an access-challenge message that includes a portion of the access list, formatted as described previously, and a State attribute (IETF RADIUS attribute 24), which includes control data used by Cisco Secure ACS to track the progress of the download. Cisco Secure ACS fits as many complete attribute-value pairs into the cisco-av-pair RADIUS VSA as it can without exceeding the maximum RADIUS message size.

The ASASM stores the portion of the access list received and responds with another access-request message that includes the same attributes as the first request for the downloadable access list, plus a copy of the State attribute received in the access-challenge message.

This process repeats until Cisco Secure ACS sends the last of the access list in an access-accept message.

### Configuring Cisco Secure ACS for Downloadable Access Lists

You can configure downloadable access lists on Cisco Secure ACS as a shared profile component and then assign the access list to a group or to an individual user.

The access list definition consists of one or more ASASM commands that are similar to the extended `access-list` command (see command reference), except without the following prefix:

```
access-list acl_name extended
```

The following example is a downloadable access list definition on Cisco Secure ACS version 3.3:

```
+--------------------------------------------+
| Shared profile Components                  |
|                                            |
| Downloadable IP ACLs Content               |
|                                            |
| Name:    acs_ten_acl                      |
|                                            |
|      ACL Definitions                       |
|                                            |
| permit tcp any host 10.0.0.254             |
| permit udp any host 10.0.0.254             |
| permit icmp any host 10.0.0.254            |
| permit tcp any host 10.0.0.253             |
| permit udp any host 10.0.0.253             |
| permit icmp any host 10.0.0.253            |
| permit tcp any host 10.0.0.252             |
| permit udp any host 10.0.0.252             |
| permit icmp any host 10.0.0.252            |
| permit ip any any                          |
+--------------------------------------------+
```

For more information about creating downloadable access lists and associating them with users, see the user guide for your version of Cisco Secure ACS.

On the ASASM, the downloaded access list has the following name:

```
#ACSACL#-ip-acl_name-number
```
The acl\_name argument is the name that is defined on Cisco Secure ACS (acs\_ten\_acl in the preceding example), and number is a unique version ID generated by Cisco Secure ACS.

The downloaded access list on the ASASM consists of the following lines:

```
access-list #ACSACL#-ip-asa-acs_ten_acl-3b5385f7 permit tcp any host 10.0.0.254
access-list #ACSACL#-ip-asa-acs_ten_acl-3b5385f7 permit udp any host 10.0.0.254
access-list #ACSACL#-ip-asa-acs_ten_acl-3b5385f7 permit icmp any host 10.0.0.254
access-list #ACSACL#-ip-asa-acs_ten_acl-3b5385f7 permit tcp any host 10.0.0.253
access-list #ACSACL#-ip-asa-acs_ten_acl-3b5385f7 permit udp any host 10.0.0.253
access-list #ACSACL#-ip-asa-acs_ten_acl-3b5385f7 permit icmp any host 10.0.0.253
access-list #ACSACL#-ip-asa-acs_ten_acl-3b5385f7 permit tcp any host 10.0.0.252
access-list #ACSACL#-ip-asa-acs_ten_acl-3b5385f7 permit udp any host 10.0.0.252
access-list #ACSACL#-ip-asa-acs_ten_acl-3b5385f7 permit icmp any host 10.0.0.252
access-list #ACSACL#-ip-asa-acs_ten_acl-3b5385f7 permit ip any any
```

### Configuring Any RADIUS Server for Downloadable Access Lists

You can configure any RADIUS server that supports Cisco IOS RADIUS VSAs to send user-specific access lists to the ASASM in a Cisco IOS RADIUS cisco-av-pair VSA (vendor 9, attribute 1).

In the cisco-av-pair VSA, configure one or more ACEs that are similar to the `access-list extended` command (see command reference), except that you replace the following command prefix:

```
access-list acl\_name extended
```

with the following text:

```
ip:inacl#nnn
```

The nnn argument is a number in the range from 0 to 999999999 that identifies the order of the command statement to be configured on the ASASM. If this parameter is omitted, the sequence value is 0, and the order of the ACEs inside the cisco-av-pair RADIUS VSA is used.

The following example is an access list definition as it should be configured for a cisco-av-pair VSA on a RADIUS server:

```
ip:inacl#1=permit tcp 10.1.0.0 255.0.0.0 10.0.0.0 255.0.0.0
ip:inacl#99=deny tcp any any
ip:inacl#2=permit udp 10.1.0.0 255.0.0.0 10.0.0.0 255.0.0.0
ip:inacl#100=deny udp any any
ip:inacl#3=permit icmp 10.1.0.0 255.0.0.0 10.0.0.0 255.0.0.0
```

For information about making unique per user the access lists that are sent in the cisco-av-pair attribute, see the documentation for your RADIUS server.

On the ASASM, the downloaded access list name has the following format:

```
AAA-user-username
```

The username argument is the name of the user that is being authenticated.

The downloaded access list on the ASASM consists of the following lines. Notice the order based on the numbers identified on the RADIUS server.

```
access-list AAA-user-bcham34-79AD4A08 permit tcp 10.1.0.0 255.0.0.0 10.0.0.0 255.0.0.0
access-list AAA-user-bcham34-79AD4A08 permit udp 10.1.0.0 255.0.0.0 10.0.0.0 255.0.0.0
access-list AAA-user-bcham34-79AD4A08 permit icmp 10.1.0.0 255.0.0.0 10.0.0.0 255.0.0.0
access-list AAA-user-bcham34-79AD4A08 deny tcp any any
access-list AAA-user-bcham34-79AD4A08 deny udp any any
```
Configuring Accounting for Network Access

Downloaded access lists have two spaces between the word “access-list” and the name. These spaces serve to differentiate a downloaded access list from a local access list. In this example, “79AD4A08” is a hash value generated by the ASASM to help determine when access list definitions have changed on the RADIUS server.

Converting Wildcard Netmask Expressions in Downloadable Access Lists

If a RADIUS server provides downloadable access lists to Cisco VPN 3000 series concentrators as well as to the ASASM, you may need the ASASM to convert wildcard netmask expressions to standard netmask expressions. This is because Cisco VPN 3000 series concentrators support wildcard netmask expressions, but the ASASM only supports standard netmask expressions. Configuring the ASASM to convert wildcard netmask expressions helps minimize the effects of these differences on how you configure downloadable access lists on your RADIUS servers. Translation of wildcard netmask expressions means that downloadable access lists written for Cisco VPN 3000 series concentrators can be used by the ASASM without altering the configuration of the downloadable access lists on the RADIUS server.

You configure access list netmask conversion on a per-server basis using the `acl-netmask-convert` command, available in the `aaa-server` configuration mode. For more information about configuring a RADIUS server, see the “Configuring AAA Server Groups” section on page 33-11. For more information about the `acl-netmask-convert` command, see the command reference.

Configuring a RADIUS Server to Download Per-User Access Control List Names

To download a name for an access list that you already created on the ASASM from the RADIUS server when a user authenticates, configure the IETF RADIUS filter-id attribute (attribute number 11) as follows:

```
filter-id=acl_name
```

Note

In Cisco Secure ACS, the values for filter-id attributes are specified in boxes in the HTML interface, omitting `filter-id=` and entering only `acl_name`.

For information about making the filter-id attribute value unique per user, see the documentation for your RADIUS server.

To create an access list on the ASASM, see Chapter 14, “Adding an Extended Access List.”

Configuring Accounting for Network Access

The ASASM can send accounting information to a RADIUS or TACACS+ server about any TCP or UDP traffic that passes through the ASASM. If that traffic is also authenticated, then the AAA server can maintain accounting information by username. If the traffic is not authenticated, the AAA server can maintain accounting information by IP address. Accounting information includes session start and stop times, username, the number of bytes that pass through the ASASM for the session, the service used, and the duration of each session.

To configure accounting, perform the following steps:
### Step 1

**Command:**

- `access-list`  

**Example:**

```
hostname(config)# access-list TELNET_AUTH extended permit tcp any any eq telnet
```

**Purpose:**

If you want the ASASM to provide accounting data per user, you must enable authentication. For more information, see the “Configuring Network Access Authentication” section on page 35-4. If you want the ASASM to provide accounting data per IP address, enabling authentication is not necessary.

The following example authenticates, authorizes, and accounts for inside Telnet traffic. Telnet traffic to servers other than 209.165.201.5 can be authenticated alone, but traffic to 209.165.201.5 requires authorization and accounting.

**Note** If you have configured authentication and want accounting data for all the traffic being authenticated, you can use the same access list that you created for use with the `aaa authentication match` command.

### Step 2

**Command:**

- `aaa accounting match acl_name interface_name server_group`  

**Example:**

```
hostname(config)# aaa accounting match SERVER_AUTH inside AuthOutBound
```

**Purpose:**

Enables accounting.

The `acl_name` argument is the access list name set in the `access-list` command.

The `interface_name` argument is the interface name set in the `nameif` command.

The `server_group` argument is the server group name set in the `aaa-server` command.

**Note** Alternatively, you can use the `aaa accounting include` command (which identifies traffic within the command), but you cannot use both methods in the same configuration. See the command reference for more information.

### Examples

The following example authenticates, authorizes, and accounts for inside Telnet traffic. Telnet traffic to servers other than 209.165.201.5 can be authenticated alone, but traffic to 209.165.201.5 requires authorization and accounting.

```
hostname(config)# aaa-server AuthOutbound protocol tacacs+
hostname(config)# exit
hostname(config)# aaa-server AuthOutbound (inside) host 10.1.1.1
hostname(config)# key TACPlusUauthKey
hostname(config)# exit
hostname(config)# access-list TELNET_AUTH extended permit tcp any any eq telnet
hostname(config)# access-list SERVER_AUTH extended permit tcp any host 209.165.201.5 eq telnet
hostname(config)# aaa authentication match TELNET_AUTH inside AuthOutBound
hostname(config)# aaa authorization match SERVER_AUTH inside AuthOutBound
hostname(config)# aaa accounting match SERVER_AUTH inside AuthOutBound
```
Using MAC Addresses to Exempt Traffic from Authentication and Authorization

The ASASM can exempt from authentication and authorization any traffic from specific MAC addresses. For example, if the ASASM authenticates TCP traffic originating on a particular network, but you want to allow unauthenticated TCP connections from a specific server, you would use a MAC exempt rule to exempt from authentication and authorization any traffic from the server specified by the rule.

This feature is particularly useful to exempt devices such as IP phones that cannot respond to authentication prompts.

To use MAC addresses to exempt traffic from authentication and authorization, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td><strong>mac-list</strong> id (deny</td>
</tr>
<tr>
<td></td>
<td>Configures a MAC list. The <em>id</em> argument is the hexadecimal number that you assign to the MAC list. To group a set of MAC addresses, enter the <strong>mac-list</strong> command as many times as needed with the same ID value. Because you can only use one MAC list for AAA exemption, be sure that your MAC list includes all the MAC addresses that you want to exempt. You can create multiple MAC lists, but you can only use one at a time. The order of entries matters, because the packet uses the first entry it matches, instead of a best match scenario. If you have a <strong>permit</strong> entry, and you want to deny an address that is allowed by the <strong>permit</strong> entry, be sure to enter the <strong>deny</strong> entry before the <strong>permit</strong> entry. The <em>mac</em> argument specifies the source MAC address in 12-digit hexadecimal form; that is, nnnnnnnnnnnnnnn. The <em>macmask</em> argument specifies the portion of the MAC address that should be used for matching. For example, ffff.ffff.ffff matches the MAC address exactly. ffff.ffff.0000 matches only the first 8 digits.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# mac-list abc permit 00a0.c95d.0282 ffff.ffff.ffff</td>
</tr>
<tr>
<td>Step 2</td>
<td><strong>aaa mac-exempt match</strong> id</td>
</tr>
<tr>
<td></td>
<td>Exempts traffic for the MAC addresses specified in a particular MAC list. The <em>id</em> argument is the string identifying the MAC list that includes the MAC addresses whose traffic is to be exempt from authentication and authorization. You can only enter one instance of the <strong>aaa mac-exempt match</strong> command.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# aaa mac-exempt match 1</td>
</tr>
</tbody>
</table>
Examples

The following example bypasses authentication for a single MAC address:

```plaintext
hostname(config)# mac-list abc permit 00a0.c95d.0282 ffff.ffff.ffff
hostname(config)# aaa mac-exempt match abc
```

The following example bypasses authentication for all Cisco IP Phones, which have the hardware ID 0003.E3:

```plaintext
hostname(config)# mac-list acd permit 0003.E300.0000 FFFF.FF00.0000
hostname(config)# aaa mac-exempt match acd
```

The following example bypasses authentication for a group of MAC addresses except for 00a0.c95d.02b2. Enter the `deny` statement before the `permit` statement, because 00a0.c95d.02b2 matches the `permit` statement as well, and if it is first, the `deny` statement will never be matched.

```plaintext
hostname(config)# mac-list 1 deny 00a0.c95d.0282 ffff.ffff.ffff
hostname(config)# mac-list 1 permit 00a0.c95d.0000 ffff.ffff.0000
hostname(config)# aaa mac-exempt match 1
```

Feature History for AAA Rules

Table 35-1 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>AAA Rules</td>
<td>7.0(1)</td>
<td>AAA Rules describe how to enable AAA for network access. We introduced the following commands: <code>aaa authentication match</code>, `aaa authentication include</td>
</tr>
</tbody>
</table>
Configuring Filtering Services

This chapter describes how to use filtering services to provide greater control over traffic passing through the ASASM and includes the following sections:

- Information About Web Traffic Filtering, page 36-1
- Configuring ActiveX Filtering, page 36-2
- Configuring Java Applet Filtering, page 36-4
- Filtering URLs and FTP Requests with an External Server, page 36-6
- Monitoring Filtering Statistics, page 36-15

Information About Web Traffic Filtering

You can use web traffic filtering in two distinct ways:

- Filtering ActiveX objects or Java applets
- Filtering with an external filtering server

Instead of blocking access altogether, you can remove specific undesirable objects from web traffic, such as ActiveX objects or Java applets, that may pose a security threat in certain situations.

You can use web traffic filtering to direct specific traffic to an external filtering server, such as Secure Computing SmartFilter (formerly N2H2) or the Websense filtering server. You can enable long URL, HTTPS, and FTP filtering using either Websense or Secure Computing SmartFilter for web traffic filtering. Filtering servers can block traffic to specific sites or types of sites, as specified by the security policy.

Note

URL caching will only work if the version of the URL server software from the URL server vendor supports it.

Because web traffic filtering is CPU-intensive, using an external filtering server ensures that the throughput of other traffic is not affected. However, depending on the speed of your network and the capacity of your web traffic filtering server, the time required for the initial connection may be noticeably slower when filtering traffic with an external filtering server.
Configuring ActiveX Filtering

This section includes the following topics:

- Information About ActiveX Filtering, page 36-2
- Licensing Requirements for ActiveX Filtering, page 36-2
- Guidelines and Limitations for ActiveX Filtering, page 36-3
- Configuring ActiveX Filtering, page 36-3
- Configuration Examples for ActiveX Filtering, page 36-3
- Feature History for ActiveX Filtering, page 36-4

Information About ActiveX Filtering

ActiveX objects may pose security risks because they can contain code intended to attack hosts and servers on a protected network. You can disable ActiveX objects with ActiveX filtering.

ActiveX controls, formerly known as OLE or OCX controls, are components that you can insert in a web page or another application. These controls include custom forms, calendars, or any of the extensive third-party forms for gathering or displaying information. As a technology, ActiveX creates many potential problems for network clients including causing workstations to fail, introducing network security problems, or being used to attack servers.

The `filteractivex` command blocks the HTML `object` commands by commenting them out within the HTML web page. ActiveX filtering of HTML files is performed by selectively replacing the `<APPLET>` and `</APPLET>`, and `<OBJECT CLASSID>` and `</OBJECT>` tags with comments. Filtering of nested tags is supported by converting top-level tags to comments.

Caution

The `filteractivex` command also blocks any Java applets, image files, or multimedia objects that are embedded in object tags.

If the `<object>` or `</object>` HTML tags split across network packets or if the code in the tags is longer than the number of bytes in the MTU, the ASASM cannot block the tag.

ActiveX blocking does not occur when users access an IP address referenced by the `alias` command or for clientless SSL VPN traffic.

Licensing Requirements for ActiveX Filtering

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>
Guidelines and Limitations for ActiveX Filtering

This section includes the guidelines and limitations for this feature.

**Context Mode Guidelines**
Supported in single and multiple context mode.

**Firewall Mode Guidelines**
Supported in routed and transparent firewall mode.

**IPv6 Guidelines**
Does not support IPv6.

Configuring ActiveX Filtering

To remove ActiveX objects in HTTP traffic that is passing through the ASASM, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>filter activex port[-port] local_ip local_mask foreign_ip foreign_mask</code></td>
<td>Removes ActiveX objects. To use this command, replace <code>port[-port]</code> with the TCP port to which filtering is applied. Typically, this is port 80, but other values are accepted. The <code>http</code> or <code>url</code> literal can be used for port 80. You can specify a range of ports by using a hyphen between the starting port number and the ending port number. The local IP address and mask identify one or more internal hosts that are the source of the traffic to be filtered. The foreign address and mask specify the external destination of the traffic to be filtered.</td>
</tr>
</tbody>
</table>

**Example:**
```
hostname# filter activex 80 0 0 0
```

Configuration Examples for ActiveX Filtering

You can set either address to `0.0.0.0` (or in shortened form, `0`) to specify all hosts. You can use `0.0.0.0` for either mask (or in shortened form, `0`) to specify all masks. This command specifies that the ActiveX object blocking applies to HTTP traffic on port 80 from any local host and for connections to any foreign host.

The following example shows how to configure ActiveX filtering to block all outbound connections:
```
hostname(config)# filter activex 80 0 0 0
```

The following example shows how to remove ActiveX filtering:
```
hostname(config)# no filter activex 80 0 0 0
```
Feature History for ActiveX Filtering

Table 36-1 lists the release history for ActiveX Filtering. ASDM is backwards-compatible with multiple platform releases, so the specific ASDM release in which support was added is not listed.

### Table 36-1  Feature History for ActiveX Filtering

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>ActiveX filtering</td>
<td>7.0(1)</td>
<td>Filters specific undesirable objects from HTTP traffic, such as ActiveX objects, which may pose a security threat in certain situations.</td>
</tr>
</tbody>
</table>

Configuring Java Applet Filtering

This section includes the following topics:

- Information About Java Applet Filtering, page 36-4
- Licensing Requirements for Java Applet Filtering, page 36-4
- Guidelines and Limitations for Java Applet Filtering, page 36-5
- Configuring Java Applet Filtering, page 36-5
- Configuration Examples for Java Applet Filtering, page 36-5
- Feature History for Java Applet Filtering, page 36-6

Information About Java Applet Filtering

Java applets may pose security risks because they can contain code intended to attack hosts and servers on a protected network. You can remove Java applets with the `filter java` command.

**Note**

Use the `filter activex` command to remove Java applets that are embedded in `<object>` tags.

The `filter java` command filters out Java applets that return to the ASASM from an outbound connection. You still receive the HTML page, but the web page source for the applet is commented out so that the applet cannot execute. The `filter java` command does not filter clientless SSL VPN traffic.

Licensing Requirements for Java Applet Filtering

The following table shows the licensing requirements for Java applet filtering:

### Table 36-2  Licensing Requirements

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>
Guidelines and Limitations for Java Applet Filtering

This section includes the guidelines and limitations for this feature.

**Context Mode Guidelines**
Supported in single and multiple context mode.

**Firewall Mode Guidelines**
Supported in routed and transparent firewall mode.

**IPv6 Guidelines**
Does not support IPv6.

Configuring Java Applet Filtering

To apply filtering to remove Java applets from HTTP traffic passing through the ASASM, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>filter java [-port] local_ip local_mask foreign_ip foreign_mask</code></td>
<td>Removes Java applets in HTTP traffic passing through the ASASM. To use this command, replace <code>[-port]</code> with the TCP port to which filtering is applied. Typically, this is port 80, but other values are accepted. The http or url literal can be used for port 80. You can specify a range of ports by using a hyphen between the starting port number and the ending port number. The local IP address and mask identify one or more internal hosts that are the source of the traffic to be filtered. The foreign address and mask specify the external destination of the traffic to be filtered. You can set either address to <code>0.0.0.0</code> (or in shortened form, <code>0</code>) to specify all hosts. You can use <code>0.0.0.0</code> for either mask (or in shortened form, <code>0</code>) to specify all hosts. You can set either address to <code>0.0.0.0</code> (or in shortened form, <code>0</code>) to specify all hosts. You can use <code>0.0.0.0</code> for either mask (or in shortened form, <code>0</code>) to specify all hosts.</td>
</tr>
</tbody>
</table>

Example:

```
hostname# filter java 80 0 0 0 0
```

Command Purpose

Configuration Examples for Java Applet Filtering

The following example specifies that Java applets are blocked on all outbound connections:

```
hostname(config)# filter java 80 0 0 0 0
```

This command specifies that the Java applet blocking applies to web traffic on port 80 from any local host and for connections to any foreign host.

The following example blocks downloading of Java applets to a host on a protected network:

```
hostname(config)# filter java http 192.168.3.3 255.255.255.255 0 0
```

This command prevents host 192.168.3.3 from downloading Java applets.
The following example removes the configuration for downloading Java applets to a host on a protected network:

```
hostname(config)# no filter java http 192.168.3.3 255.255.255.255 0 0
```

This command allows host 192.168.3.3 to download Java applets.

## Feature History for Java Applet Filtering

Table 36-1 lists the release history for Java applet filtering. ASDM is backwards-compatible with multiple platform releases, so the specific ASDM release in which support was added is not listed.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Java applet filtering</td>
<td>7.0(1)</td>
<td>Filters specific undesirable objects from HTTP traffic, such as Java applets, which may pose a security threat in certain situations.</td>
</tr>
</tbody>
</table>

## Filtering URLs and FTP Requests with an External Server

This section describes how to filter URLs and FTP requests with an external server and includes the following topics:

- Information About URL Filtering, page 36-6
- Licensing Requirements for URL Filtering, page 36-7
- Guidelines and Limitations for URL Filtering, page 36-7
- Identifying the Filtering Server, page 36-8
- Configuring Additional URL Filtering Settings, page 36-10
- Feature History for URL Filtering, page 36-17

## Information About URL Filtering

You can apply filtering to connection requests originating from a more secure network to a less secure network. Although you can use ACLs to prevent outbound access to specific content servers, managing usage this way is difficult because of the size and dynamic nature of the Internet. You can simplify configuration and improve ASASM performance by using a separate server running one of the following Internet filtering products:

- Websense Enterprise for filtering HTTP, HTTPS, and FTP.
- McAfee SmartFilter (formerly N2H2) for filtering HTTP, HTTPS, FTP, and long URL filtering.

In long URLs, the URL in the Referer field might contain a “host:” text string, which could cause the HTTP GET header to be incorrectly parsed as containing the HTTP Host parameter. The ASASM, however, correctly parses the Referer field even when it contains a “host:” text string and forwards the header to the McAfee SmartFilter server with the correct Referer URL.
Note

URL caching will only work if the version of the URL server software from the URL server vendor supports it.

Although ASASM performance is less affected when using an external server, you might notice longer access times to websites or FTP servers when the filtering server is remote from the ASASM.

When filtering is enabled and a request for content is directed through the ASASM, the request is sent to the content server and to the filtering server at the same time. If the filtering server allows the connection, the ASASM forwards the response from the content server to the originating client. If the filtering server denies the connection, the ASASM drops the response and sends a message or return code indicating that the connection was not successful.

If user authentication is enabled on the ASASM, then the ASASM also sends the username to the filtering server. The filtering server can use user-specific filtering settings or provide enhanced reporting about usage.

Licensing Requirements for URL Filtering

The following table shows the licensing requirements for URL filtering:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License</td>
</tr>
</tbody>
</table>

Guidelines and Limitations for URL Filtering

This section includes the guidelines and limitations for this feature.

Context Mode Guidelines
Supported in single and multiple context mode.

Firewall Mode Guidelines
Supported in routed and transparent firewall mode.

IPv6 Guidelines
Does not support IPv6.
### Identifying the Filtering Server

You can identify up to four filtering servers per context. The ASASM uses the servers in order until a server responds. In single mode, a maximum of 16 of the same type of filtering servers are allowed. You can only configure a single type of server (Websense or Secure Computing SmartFilter) in your configuration.

**Note**

You must add the filtering server before you can configure filtering for HTTP or HTTPS with the `filter` command. If you remove the filtering servers from the configuration, then all `filter` commands are also removed.

To specify the external filtering server, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Choose from the following options:</td>
</tr>
</tbody>
</table>
### Command

<table>
<thead>
<tr>
<th>For Websense:</th>
</tr>
</thead>
<tbody>
<tr>
<td>hostname(config)# url-server (if_name) host local_ip [timeout seconds] [protocol TCP</td>
</tr>
</tbody>
</table>

**Example:**

hostname(config)# url-server (perimeter) host 10.0.1.1 protocol TCP version 4

Identifies the address of the filtering server. *if_name* is the name of the ASASM interface connected to the filtering server (the default is inside). For the *vendor* option, use *secure-computing* as the vendor string; however, *n2h2* is acceptable for backward compatibility. When the configuration entries are generated, *secure-computing* is saved as the vendor string. The *host local_ip* option is the IP address of the URL filtering server. The *port number* option is the Secure Computing SmartFilter server port number of the filtering server; the ASASM also listens for UDP replies on this port.

**Note**

The default port is 4005, which is used by the Secure Computing SmartFilter server to communicate to the ASASM via TCP or UDP. For information about changing the default port, see the Filtering by N2H2 Administrator's Guide.

The *timeout seconds* option is the number of seconds that the ASASM should keep trying to connect to the filtering server. The *connections number* option is the number of tries to make a connection between the host and server.

The example identifies a Websense filtering server with the IP address 10.0.1.1 on a perimeter interface of the ASASM. Version 4, which is enabled in this example, is recommended by Websense because it supports caching.

<table>
<thead>
<tr>
<th>For Secure Computing SmartFilter (formerly N2H2):</th>
</tr>
</thead>
<tbody>
<tr>
<td>hostname(config)# url-server (if_name) vendor (secure-computing</td>
</tr>
</tbody>
</table>

**Example:**

hostname(config)# url-server (perimeter) vendor n2h2 host 10.0.1.1
hostname(config)# url-server (perimeter) vendor n2h2 host 10.0.1.2

The example identifies redundant Secure Computing SmartFilter servers that are both on a perimeter interface of the ASASM.
Configuring Additional URL Filtering Settings

After you have accessed a website, the filtering server can allow the ASASM to cache the server address for a certain period of time, as long as each website hosted at the address is in a category that is permitted at all times. When you access the server again, or if another user accesses the server, the ASASM does not need to consult the filtering server again to obtain the server address.

Note

Requests for cached IP addresses are not passed to the filtering server and are not logged. As a result, this activity does not appear in any reports.

This section describes how to configure additional URL filtering settings and includes the following topics:

- Buffering the Content Server Response, page 36-10
- Caching Server Addresses, page 36-11
- Filtering HTTP URLs, page 36-11
- Filtering HTTPS URLs, page 36-13
- Filtering FTP Requests, page 36-14

Buffering the Content Server Response

When you issue a request to connect to a content server, the ASASM sends the request to the content server and to the filtering server at the same time. If the filtering server does not respond before the content server, the server response is dropped. This behavior delays the web server response for the web client, because the web client must reissue the request.

By enabling the HTTP response buffer, replies from web content servers are buffered, and the responses are forwarded to the requesting client if the filtering server allows the connection. This behavior prevents the delay that might otherwise occur.

To configure buffering for responses to HTTP or FTP requests, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> url-block block block-buffer-limit</td>
<td>Enables buffering of responses for HTTP or FTP requests that are pending a response from the filtering server. Replaces block-buffer with the maximum number of HTTP responses that can be buffered while awaiting responses from the URL server.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname# url-block 3000</td>
<td></td>
</tr>
</tbody>
</table>

Note

Buffering of URLs longer than 3072 bytes is not supported.

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 2</strong> url-block mempool-size memory-pool-size</td>
<td>Configures the maximum memory available for buffering pending URLs (and for buffering long URLs). Replaces memory-pool-size with a value from 2 to 10240 for a maximum memory allocation of 2 KB to 10 MB.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname# url-block mempool-size 5000</td>
<td></td>
</tr>
</tbody>
</table>
Caching Server Addresses

After you access a website, the filtering server can allow the ASASM to cache the server address for a certain period of time, as long as each website hosted at the address is in a category that is permitted at all times. When you access the server again, or if another user accesses the server, the ASASM does not need to consult the filtering server again.

Note
Requests for cached IP addresses are not passed to the filtering server and are not logged. As a result, this activity does not appear in any reports. You can accumulate Websense run logs before using the url-cache command.

To improve throughput, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>url-cache dst</td>
<td>src_dst size</td>
</tr>
<tr>
<td>Example: hostname## url-cache src_dst 100</td>
<td>Uses the dst keyword to cache entries based on the URL destination address. Choose this option if all users share the same URL filtering policy on the Websense server. Uses the src_dst keyword to cache entries based on both the source address initiating the URL request as well as the URL destination address. Choose this option if users do not share the same URL filtering policy on the Websense server.</td>
</tr>
</tbody>
</table>

Filtering HTTP URLs

This section describes how to configure HTTP filtering with an external filtering server and includes the following topics:

- Enabling HTTP Filtering, page 36-12
- Enabling Filtering of Long HTTP URLs, page 36-12
- Truncating Long HTTP URLs, page 36-13
- Exempting Traffic from Filtering, page 36-13
Enabling HTTP Filtering

You must identify and enable the URL filtering server before enabling HTTP filtering. When the filtering server approves an HTTP connection request, the ASASM allows the reply from the web server to reach the originating client. If the filtering server denies the request, the ASASM redirects you to a block page, indicating that access was denied.

To enable HTTP filtering, enter the following command:

```
hostname# filter url http 80 allow proxy-block
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`filter url [http</td>
<td>port[-port] local_ip local_mask foreign_ip foreign_mask] [allow] [proxy-block]`</td>
</tr>
</tbody>
</table>

Enabling Filtering of Long HTTP URLs

By default, the ASASM considers an HTTP URL to be a long URL if it is greater than 1159 characters. You can increase the maximum length allowed.

To configure the maximum size of a single URL, enter the following command:

```
hostname# url-block url-size 3
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>url-block url-size long-url-size</code></td>
<td>Replaces the <code>long-url-size</code> with the maximum size in KB for each long URL being buffered. For Websense servers, this is a value from 2 to 4 for a maximum URL size from 2 KB to 4 KB; for Secure Computing SmartFilter servers, this is a value between 2 and 3 for a maximum URL size from 2 KB to 3 KB. The default value is 2.</td>
</tr>
</tbody>
</table>
Truncating Long HTTP URLs

By default, if a URL exceeds the maximum permitted size, then it is dropped. To avoid this occurrence, truncate a long URL by entering the following command:

```
hostname# filter url longurl-truncate
```

### Command

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>filter url [longurl-truncate</td>
<td>The <code>longurl-truncate</code> option causes the ASASM to send only the</td>
</tr>
<tr>
<td>longurl-deny</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Exempting Traffic from Filtering

To exempt traffic from filtering, enter following command:

```
hostname(config)# filter url http 0 0 0 0
hostname(config)# filter url except 10.0.2.54 255.255.255.255 0 0
```

### Command

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>filter url except source_ip</td>
<td>Exempts specific traffic from filtering.</td>
</tr>
<tr>
<td>source_mask</td>
<td>The example shows how to cause all HTTP requests to be forwarded to the</td>
</tr>
<tr>
<td>dest_ip dest_mask</td>
<td>filtering server, except for those from 10.0.2.54.</td>
</tr>
</tbody>
</table>

Filtering HTTPS URLs

You must identify and enable the URL filtering server before enabling HTTPS filtering.

**Note**

Websense and Secure Computing Smartfilter currently support HTTPS; older versions of the Secure Computing SmartFilter (formerly N2H2) do not support HTTPS filtering.

Because HTTPS content is encrypted, the ASASM sends the URL lookup without directory and filename information. When the filtering server approves an HTTPS connection request, the ASASM allows the completion of SSL connection negotiation and allows the reply from the web server to reach the originating client. If the filtering server denies the request, the ASASM prevents the completion of SSL connection negotiation. The browser displays an error message, such as “The Page or the content cannot be displayed.”

**Note**

The ASASM does not provide an authentication prompt for HTTPS, so you must authenticate with the ASASM using HTTP or FTP before accessing HTTPS servers.
To enable HTTPS filtering, enter the following command:

```
filter https port[-port] localIP [local_mask] foreign_IP foreign_mask [allow]
```

**Example:**
```
hostname# filter https 443 0 0 0 0 0 0 0 0 allow
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>filter https</code></td>
<td>Enables HTTPS filtering.</td>
</tr>
<tr>
<td><code>port[-port]</code></td>
<td>Replaces <code>port[-port]</code> with a range of port numbers if a different port than the default port for HTTPS (443) is used.</td>
</tr>
<tr>
<td><code>localIP [local_mask] foreign_IP foreign_mask [allow]</code></td>
<td>Replaces <code>local_ip</code> and <code>local_mask</code> with the IP address and subnet mask of a user or subnetwork making requests.</td>
</tr>
<tr>
<td></td>
<td>Replaces <code>foreign_ip</code> and <code>foreign_mask</code> with the IP address and subnet mask of a server or subnetwork responding to requests.</td>
</tr>
<tr>
<td></td>
<td>The <code>allow</code> option causes the ASASM to forward HTTPS traffic without filtering when the primary filtering server is unavailable.</td>
</tr>
</tbody>
</table>

**Filtering FTP Requests**

You must identify and enable the URL filtering server before enabling FTP filtering.

**Note**

Websense and Secure Computing Smartfilter currently support FTP; older versions of Secure Computing SmartFilter (formerly known as N2H2) did not support FTP filtering.

When the filtering server approves an FTP connection request, the ASASM allows the successful FTP return code to reach the originating client. For example, a successful return code is “250: CWD command successful.” If the filtering server denies the request, the FTP return code is changed to show that the connection was denied. For example, the ASASM changes code 250 to “550 Requested file is prohibited by URL filtering policy.”

To enable FTP filtering, enter the following command:

```
filter ftp port[-port] localIP [local_mask] foreign_IP foreign_mask [allow] [interact-block]
```

**Example:**
```
hostname# filter ftp 21 0 0 0 0 0 0 0 0 allow
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>filter ftp</code></td>
<td>Enables FTP filtering.</td>
</tr>
<tr>
<td><code>port[-port] localIP [local_mask] foreign_IP foreign_mask [allow] [interact-block]</code></td>
<td>Replaces <code>port[-port]</code> with a range of port numbers if a different port than the default port for FTP (21) is used.</td>
</tr>
<tr>
<td></td>
<td>Replaces <code>local_ip</code> and <code>local_mask</code> with the IP address and subnet mask of a user or subnetwork making requests.</td>
</tr>
<tr>
<td></td>
<td>Replaces <code>foreign_ip</code> and <code>foreign_mask</code> with the IP address and subnet mask of a server or subnetwork responding to requests.</td>
</tr>
<tr>
<td></td>
<td>The <code>allow</code> option causes the ASASM to forward HTTPS traffic without filtering when the primary filtering server is unavailable.</td>
</tr>
<tr>
<td></td>
<td>Use the <code>interact-block</code> option to prevent interactive FTP sessions that do not provide the entire directory path. An interactive FTP client allows you to change directories without typing the entire path. For example, you might enter <code>cd ./files</code> instead of <code>cd /public/files</code>.</td>
</tr>
</tbody>
</table>
Monitoring Filtering Statistics

To monitor filtering statistics, enter one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show url-server</td>
<td>Shows information about the URL filtering server.</td>
</tr>
<tr>
<td>show url-server statistics</td>
<td>Shows URL filtering statistics.</td>
</tr>
<tr>
<td>show url-block</td>
<td>Shows the number of packets held in the url-block buffer and the number (if any) dropped because of exceeding the buffer limit or retransmission.</td>
</tr>
<tr>
<td>show url-block block statistics</td>
<td>Shows the URL block statistics.</td>
</tr>
<tr>
<td>show url-cache stats</td>
<td>Shows the URL cache statistics.</td>
</tr>
<tr>
<td>show perfmon</td>
<td>Shows URL filtering performance statistics, along with other performance statistics.</td>
</tr>
<tr>
<td>show filter</td>
<td>Shows the filtering configuration.</td>
</tr>
</tbody>
</table>

Examples

The following is sample output from the `show url-server` command:

```
hostname# show url-server
url-server (outside) vendor n2h2 host 128.107.254.202 port 4005 timeout 5 protocol TCP
```

The following is sample output from the `show url-server statistics` command:

```
hostname# show url-server statistics

Global Statistics:
--------------------
URLs total/allowed/denied         13/3/10
URLs allowed by cache/server     0/3
URLs denied by cache/server      0/10
HTTPSs total/allowed/denied      138/137/1
HTTPSs allowed by cache/server   0/137
HTTPSs denied by cache/server    0/1
FTPs total/allowed/denied        0/0/0
FTPs allowed by cache/server     0/0
FTPs denied by cache/server      0/0
Requests dropped                 0
Server timeouts/retries          0/0
Processed rate average 60s/300s  0/0 requests/second
Denied rate average 60s/300s      0/0 requests/second
Dropped rate average 60s/300s     0/0 requests/second

Server Statistics:
--------------------
10.125.76.20                     UP
Vendor                          websense
Port                            15868
Requests total/allowed/denied   151/140/11
Server timeouts/retries         0/0
Responses received              151
Response time average 60s/300s   0/0

URL Packets Sent and Received Stats:
-------------------------------------
Message     Sent     Received
```
STATUS_REQUEST 1609 1601
LOOKUP_REQUEST 1526 1526
LOG_REQUEST 0 NA

Errors:
-------
RFC noncompliant GET method 0
URL buffer update failure 0

The following is sample output from the show url-block command:

hostname# show url-block
  url-block url-mempool 128
  url-block url-size 4
  url-block block 128

The following is sample output from the show url-block block statistics command:

hostname# show url-block block statistics
URL Pending Packet Buffer Stats with max block 128
-----------------------------------------------
Cumulative number of packets held: 896
Maximum number of packets held (per URL): 3
Current number of packets held (global): 38
Packets dropped due to exceeding url-block buffer limit: 7546
HTTP server retransmission: 10
Number of packets released back to client: 0

The following is sample output from the show url-cache stats command:

hostname# show url-cache stats
URL Filter Cache Stats
--------------------
Size : 128KB
Entries : 1724
In Use : 456
Lookups : 45
Hits : 8

This shows how the cache is used.

The following is sample output from the show perfmon command:

hostname# show perfmon
PERFMON STATS: Current Average
Xlates 0/s 0/s
Connections 0/s 2/s
TCP Conn 0/s 2/s
UDP Conn 0/s 0/s
URL Access 0/s 2/s
URL Server Req 0/s 3/s
TCP Fixup 0/s 0/s
TCP Intercept 0/s 0/s
HTTP Fixup 0/s 3/s
FTP Fixup 0/s 0/s
AAA Authen 0/s 0/s
AAA Author 0/s 0/s
AAA Account 0/s 0/s

The following is sample output from the show filter command:

hostname# show filter
filter url http 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
Feature History for URL Filtering

Table 36-5 lists the release history for URL filtering. ASDM is backwards-compatible with multiple platform releases, so the specific ASDM release in which support was added is not listed.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>URL filtering</td>
<td>7.0(1)</td>
<td>Filters URLs based on an established set of filtering criteria.</td>
</tr>
</tbody>
</table>
Configuring Web Cache Services Using WCCP

This chapter describes how to configure web caching services using WCCP, and includes the following sections:

- Information About WCCP, page 37-1
- Guidelines and Limitations, page 37-1
- Licensing Requirements for WCCP, page 37-2
- Enabling WCCP Redirection, page 37-3
- WCCP Monitoring Commands, page 37-4
- Feature History for WCCP, page 37-4

Information About WCCP

The purpose of web caching is to reduce latency and network traffic. Previously-accessed web pages are stored in a cache buffer, so if users need the page again, they can retrieve it from the cache instead of the web server.

WCCP specifies interactions between the ASASM and external web caches. The feature transparently redirects selected types of traffic to a group of web cache engines to optimize resource usage and lower response times. The ASASM only supports WCCP Version 2.

Using an ASASM as an intermediary eliminates the need for a separate router to do the WCCP redirection, because the ASASM redirects requests to cache engines. When the ASASM determines that a packet needs redirection, it skips TCP state tracking, TCP sequence number randomization, and NAT on these traffic flows.

Guidelines and Limitations

The following WCCPv2 features are supported for the ASASM:

- Redirection of multiple TCP and UDP port-destined traffic.
- Authentication for cache engines in a service group.
- Multiple cache engines in a service group.
- GRE encapsulation.

The following WCCPv2 features are not supported for the ASASM:
Licensing Requirements for WCCP

Table 37-1  Licensing Requirements

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>
### Enabling WCCP Redirection

**Note**

The ASA selects the highest IP address configured on any interface as the WCCP router ID. This address is used to establish a GRE tunnel with the cache engine.

WCCP redirection is supported only on the ingress of an interface. The only topology that the ASASM supports is when client and cache engine are behind the same interface of the ASASM and the cache engine can directly communicate with the client, without going through the ASASM.

The following configuration tasks assume you have already installed and configured the cache engines that you want to include in your network.

To configure WCCP redirection, perform the following steps:

<table>
<thead>
<tr>
<th>Step 1</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>wccp (\text{web-cache}</td>
<td>service_number) [redirect-list access_list] [group-list access_list] [password password]</td>
<td>Enables a WCCP service group and identifies the service to be redirected. (Optional) Also defines which cache engines can participate in the service group, and what traffic should be redirected to the cache engine. The standard service is web-cache, which intercepts TCP port 80 (HTTP) traffic and redirects that traffic to the cache engines, but you can identify a service number (if desired) between 0 and 254. For example, to transparently redirect native FTP traffic to a cache engine, use WCCP service 60. You can enter this command multiple times for each service group that you want to enable. The <strong>redirect-list access_list</strong> argument controls traffic that is redirected to this service group. The <strong>group-list access_list</strong> argument determines which web cache IP addresses are allowed to participate in the service group. The <strong>password password</strong> argument specifies MD5 authentication for messages that are received from the service group. Messages that are not accepted by the authentication are discarded.</td>
</tr>
<tr>
<td>hostname (config)# wccp web-cache</td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname (config)# wccp interface inside web-cache redirect in</td>
<td>Identifies an interface and enables WCCP redirection on the interface. The standard service is web-cache, which intercepts TCP port 80 (HTTP) traffic and redirects that traffic to the cache engines, but you can identify a service number (if desired) between 0 and 254. For example, to transparently redirect native FTP traffic to a cache engine, use WCCP service 60. You can enter this command multiple times for each service group that you want to enable.</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>hostname (config)# wccp interface inside web-cache redirect in</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Examples**

For example, to enable the standard web-cache service and redirect HTTP traffic that enters the inside interface to a web cache, enter the following commands:

```
hostname (config)# wccp web-cache
hostname (config)# wccp interface inside web-cache redirect in
```
**WCCP Monitoring Commands**

To monitor WCCP, enter one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show running-config wccp</td>
<td>Shows the current WCCP configuration.</td>
</tr>
<tr>
<td>show running-config wccp interface</td>
<td>Shows the current WCCP interfaces status.</td>
</tr>
</tbody>
</table>

**Feature History for WCCP**

Table 37-2 lists the release history for this feature.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>WCCP</td>
<td>7.2(1)</td>
<td>WCCP specifies interactions between the ASASM and external web caches. We introduced the following commands: wccp and wccp interface</td>
</tr>
</tbody>
</table>
Configuring Digital Certificates

This chapter describes how to configure digital certificates and includes the following sections:

- Information About Digital Certificates, page 38-1
- Licensing Requirements for Digital Certificates, page 38-7
- Prerequisites for Local Certificates, page 38-7
- Guidelines and Limitations, page 38-8
- Configuring Digital Certificates, page 38-9
- Monitoring Digital Certificates, page 38-41
- Feature History for Certificate Management, page 38-43

Information About Digital Certificates

CAs are responsible for managing certificate requests and issuing digital certificates. A digital certificate includes information that identifies a user or device, such as a name, serial number, company, department, or IP address. A digital certificate also includes a copy of the public key for the user or device. A CA can be a trusted third party, such as VeriSign, or a private (in-house) CA that you establish within your organization.

For an example of a scenario that includes certificate configuration and load balancing, see the following URL: https://supportforums.cisco.com/docs/DOC-5964.

This section includes the following topics:

- Public Key Cryptography, page 38-2
- Certificate Scalability, page 38-2
- Key Pairs, page 38-2
- Trustpoints, page 38-3
- Revocation Checking, page 38-4
- The Local CA, page 38-6
Public Key Cryptography

Digital signatures, enabled by public key cryptography, provide a way to authenticate devices and users. In public key cryptography, such as the RSA encryption system, each user has a key pair containing both a public and a private key. The keys act as complements, and anything encrypted with one of the keys can be decrypted with the other.

In simple terms, a signature is formed when data is encrypted with a private key. The signature is attached to the data and sent to the receiver. The receiver applies the public key of the sender to the data. If the signature sent with the data matches the result of applying the public key to the data, the validity of the message is established.

This process relies on the receiver having a copy of the public key of the sender and a high degree of certainty that this key belongs to the sender, not to someone pretending to be the sender.

Obtaining the public key of a sender is normally handled externally or through an operation performed at installation. For example, most web browsers are configured with the root certificates of several CAs by default. For VPN, the IKE protocol, a component of IPsec, can use digital signatures to authenticate peer devices before setting up security associations.

Certificate Scalability

Without digital certificates, you must manually configure each IPsec peer for each peer with which it communicates; as a result, each new peer that you add to a network would require a configuration change on each peer with which it needs to communicate securely.

When you use digital certificates, each peer is enrolled with a CA. When two peers try to communicate, they exchange certificates and digitally sign data to authenticate each other. When a new peer is added to the network, you enroll that peer with a CA and none of the other peers need modification. When the new peer attempts an IPsec connection, certificates are automatically exchanged and the peer can be authenticated.

With a CA, a peer authenticates itself to the remote peer by sending a certificate to the remote peer and performing some public key cryptography. Each peer sends its unique certificate, which was issued by the CA. This process works because each certificate encapsulates the public key for the associated peer, each certificate is authenticated by the CA, and all participating peers recognize the CA as an authenticating authority. The process is called IKE with an RSA signature.

The peer can continue sending its certificate for multiple IPsec sessions, and to multiple IPsec peers, until the certificate expires. When its certificate expires, the peer administrator must obtain a new one from the CA.

CAs can also revoke certificates for peers that no longer participate in IPsec. Revoked certificates are not recognized as valid by other peers. Revoked certificates are listed in a CRL, which each peer may check before accepting a certificate from another peer.

Some CAs have an RA as part of their implementation. An RA is a server that acts as a proxy for the CA, so that CA functions can continue when the CA is unavailable.

Key Pairs

Key pairs are RSA keys, which have the following characteristics:

- RSA keys can be used for SSH or SSL.
- SCEP enrollment supports the certification of RSA keys.
For the purposes of generating keys, the maximum key modulus for RSA keys is 2048 bits. The default size is 1024. Many SSL connections using identity certificates with RSA key pairs that exceed 1024 bits can cause a high CPU usage on the ASASM and rejected clientless logins.

For signature operations, the supported maximum key size is 4096 bits.

You can generate a general purpose RSA key pair, used for both signing and encryption, or you can generate separate RSA key pairs for each purpose. Separate signing and encryption keys help to reduce exposure of the keys, because SSL uses a key for encryption but not signing. However, IKE uses a key for signing but not encryption. By using separate keys for each, exposure of the keys is minimized.

Trustpoints

Trustpoints let you manage and track CAs and certificates. A trustpoint is a representation of a CA or identity pair. A trustpoint includes the identity of the CA, CA-specific configuration parameters, and an association with one, enrolled identity certificate.

After you have defined a trustpoint, you can reference it by name in commands requiring that you specify a CA. You can configure many trustpoints.

Note

If an ASASM has multiple trustpoints that share the same CA, only one of these trustpoints sharing the CA can be used to validate user certificates. To control which trustpoint sharing a CA is used for validation of user certificates issued by that CA, use the `support-user-cert-validation` command.

For automatic enrollment, a trustpoint must be configured with an enrollment URL, and the CA that the trustpoint represents must be available on the network and must support SCEP.

You can export and import the keypair and issued certificates associated with a trustpoint in PKCS12 format. This format is useful to manually duplicate a trustpoint configuration on a different ASASM.

Certificate Enrollment

The ASASM needs a CA certificate for each trustpoint and one or two certificates for itself, depending upon the configuration of the keys used by the trustpoint. If the trustpoint uses separate RSA keys for signing and encryption, the ASASM needs two certificates, one for each purpose. In other key configurations, only one certificate is needed.

The ASASM supports automatic enrollment with SCEP and with manual enrollment, which lets you paste a base-64-encoded certificate directly into the terminal. For site-to-site VPNs, you must enroll each ASASM. For remote access VPNs, you must enroll each ASASM and each remote access VPN client.

Proxy for SCEP Requests

The ASASM can proxy SCEP requests between AnyConnect and a third-party CA. The CA only needs to be accessible to the ASA if it is acting as the proxy. For the ASASM to provide this service, the user must authenticate using any of the methods supported by AAA before the ASA sends an enrollment request. You can also use host scan and dynamic access policies to enforce rules of eligibility to enroll.

The ASASM supports this feature only with an AnyConnect SSL or IKEv2 VPN session. It supports all SCEP-compliant CAs, including IOS CS, Windows Server 2003 CA, and Windows Server 2008 CA.

Clientless (browser-based) access does not support SCEP proxy, although WebLaunch—clientless-initiated AnyConnect—does support it.
Information About Digital Certificates

The ASASM does not support polling for certificates.
The ASASM supports load balancing for this feature.

Revocation Checking

When a certificate is issued, it is valid for a fixed period of time. Sometimes a CA revokes a certificate before this time period expires; for example, because of security concerns or a change of name or association. CAs periodically issue a signed list of revoked certificates. Enabling revocation checking forces the ASASM to check that the CA has not revoked a certificate each time that it uses the certificate for authentication.

When you enable revocation checking, the ASASM checks certificate revocation status during the PKI certificate validation process, which can use either CRL checking, OCSP, or both. OCSP is only used when the first method returns an error (for example, indicating that the server is unavailable).

With CRL checking, the ASASM retrieves, parses, and caches CRLs, which provide a complete list of revoked (and unrevoked) certificates with their certificate serial numbers. The ASASM evaluates certificates according to CRLs, also called authority revocation lists, from the identity certificate up the chain of subordinate certificate authorities.

OCSP offers a more scalable method of checking revocation status in that it localizes certificate status through a validation authority, which it queries for status of a specific certificate.

Supported CA Servers

The ASASM supports the following CA servers:
Cisco IOS CS, ASASM Local CA, and third-party X.509 compliant CA vendors including, but not limited to:
- Baltimore Technologies
- Entrust
- Digicert
- Geotrust
- GoDaddy
- iPlanet/Netscape
- Microsoft Certificate Services
- RSA Keon
- Thawte
- VeriSign

CRLs

CRLs provide the ASASM with one way of determining whether a certificate that is within its valid time range has been revoked by the issuing CA. CRL configuration is part of configuration of a trustpoint.

You can configure the ASASM to make CRL checks mandatory when authenticating a certificate by using the `revocation-check crl` command. You can also make the CRL check optional by using the `revocation-check crl none` command, which allows the certificate authentication to succeed when the CA is unavailable to provide updated CRL data.
The ASASM can retrieve CRLs from CAs using HTTP, SCEP, or LDAP. CRLs retrieved for each trustpoint are cached for a configurable amount of time for each trustpoint.

When the ASASM has cached a CRL for longer than the amount of time it is configured to cache CRLs, the ASASM considers the CRL too old to be reliable, or “stale.” The ASASM tries to retrieve a newer version of the CRL the next time that a certificate authentication requires a check of the stale CRL.

The ASASM caches CRLs for an amount of time determined by the following two factors:

- The number of minutes specified with the `cache-time` command. The default value is 60 minutes.
- The NextUpdate field in the CRLs retrieved, which may be absent from CRLs. You control whether the ASASM requires and uses the NextUpdate field with the `enforcenextupdate` command.

The ASASM uses these two factors in the following ways:

- If the NextUpdate field is not required, the ASASM marks CRLs as stale after the length of time defined by the `cache-time` command.
- If the NextUpdate field is required, the ASASM marks CRLs as stale at the sooner of the two times specified by the `cache-time` command and the NextUpdate field. For example, if the `cache-time` command is set to 100 minutes and the NextUpdate field specifies that the next update is 70 minutes away, the ASASM marks CRLs as stale in 70 minutes.

If the ASASM has insufficient memory to store all CRLs cached for a given trustpoint, it deletes the least recently used CRL to make room for a newly retrieved CRL.

**OCSP**

OCSP provides the ASASM with a way of determining whether a certificate that is within its valid time range has been revoked by the issuing CA. OCSP configuration is part of trustpoint configuration.

OCSP localizes certificate status on a validation authority (an OCSP server, also called the `responder`) which the ASASM queries for the status of a specific certificate. This method provides better scalability and more up-to-date revocation status than does CRL checking, and helps organizations with large PKI installations deploy and expand secure networks.

---

**Note**

The ASASM allows a five-second time skew for OCSP responses.

You can configure the ASASM to make OCSP checks mandatory when authenticating a certificate by using the `revocation-check ocsp` command. You can also make the OCSP check optional by using the `revocation-check ocsp none` command, which allows the certificate authentication to succeed when the validation authority is unavailable to provide updated OCSP data.

OCSP provides three ways to define the OCSP server URL. The ASASM uses these servers in the following order:

1. The OCSP URL defined in a match certificate override rule by using the `match certificate` command.
2. The OCSP URL configured by using the `ocsp url` command.
3. The AIA field of the client certificate.

---

**Note**

To configure a trustpoint to validate a self-signed OCSP responder certificate, you import the self-signed responder certificate into its own trustpoint as a trusted CA certificate. Then you configure the `match certificate` command in the client certificate validating trustpoint to use the trustpoint that includes the self-signed OCSP responder certificate to validate the responder certificate. Use the same procedure for
configuring validating responder certificates external to the validation path of the client certificate.

The OCSP server (responder) certificate usually signs the OCSP response. After receiving the response, the ASASM tries to verify the responder certificate. The CA normally sets the lifetime of the OCSP responder certificate to a relatively short period to minimize the chance of being compromised. The CA usually also includes an ocsp-no-check extension in the responder certificate, which indicates that this certificate does not need revocation status checking. However, if this extension is not present, the ASASM tries to check revocation status using the same method specified in the trustpoint. If the responder certificate is not verifiable, revocation checks fail. To avoid this possibility, use the `revocation-check none` command to configure the responder certificate validating trustpoint, and use the `revocation-check ocsp` command to configure the client certificate.

### The Local CA

The local CA performs the following tasks:

- Integrates basic certificate authority operation on the ASASM.
- Deploys certificates.
- Provides secure revocation checking of issued certificates.
- Provides a certificate authority on the ASASM for use with browser-based and client-based SSL VPN connections.
- Provides trusted digital certificates to users, without the need to rely on external certificate authorization.
- Provides a secure, in-house authority for certificate authentication and offers straightforward user enrollment by means of a website login.

### Storage for Local CA Files

The ASASM accesses and implements user information, issued certificates, and revocation lists using a local CA database. This database resides in local flash memory by default, or can be configured to reside on an external file system that is mounted and accessible to the ASASM.

No limits exist on the number of users that can be stored in the local CA user database; however, if flash memory storage issues arise, syslogs are generated to alert the administrator to take action, and the local CA could be disabled until the storage issues are resolved. Flash memory can store a database with 3500 users or less; however, a database of more than 3500 users requires external storage.

### The Local CA Server

After you configure a local CA server on the ASASM, users can enroll for a certificate by logging into a website and entering a username and a one-time password that is provided by the local CA administrator to validate their eligibility for enrollment.

As shown in Figure 38-1, the local CA server resides on the ASASM and handles enrollment requests from website users and CRL inquiries coming from other certificate validating devices and ASASMs. Local CA database and configuration files are maintained either on the ASASM flash memory (default storage) or on a separate storage device.
Licensing Requirements for Digital Certificates

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

Prerequisites for Local Certificates

Local certificates have the following prerequisites:

- Make sure that the ASASM is configured correctly to support certificates. An incorrectly configured ASASM can cause enrollment to fail or request a certificate that includes inaccurate information.
- Make sure that the hostname and domain name of the ASASM are configured correctly. To view the currently configured hostname and domain name, enter the `show running-config` command. For information about configuring the hostname and domain name, see the “Configuring the Hostname, Domain Name, and Passwords” section on page 9-1.
- Make sure that the ASASM clock is set accurately before configuring the CA. Certificates have a date and time that they become valid and expire. When the ASASM enrolls with a CA and obtains a certificate, the ASASM checks that the current time is within the valid range for the certificate. If it is outside that range, enrollment fails.

Prerequisites for SCEP Proxy Support

Configuring the ASA as a proxy to submit requests for third-party certificates has the following requirements:

- AnyConnect Secure Mobility Client 3.0 or later must be running at the endpoint.
- The authentication method, configured in the connection profile for your group policy, must be set to use both AAA and certificate authentication.
• An SSL port must be open for IKEv2 VPN connections.
• The CA must be in auto-grant mode.

Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

Context Mode Guidelines
• Supported in single and multiple context mode for a local CA.
• Supported in single context mode only for third-party CAs.

Firewall Mode Guidelines
Supported in routed and transparent firewall mode.

Failover Guidelines
• Does not support replicating sessions in Stateful Failover.
• Does not support Active/Active or Active/Standby failover.

IPv6 Guidelines
Supports IPv6.

Additional Guidelines
• For ASASMs that are configured as CA servers or clients, limit the validity period of the certificate to less than the recommended end date of 03:14:08 UTC, January 19, 2038. This guideline also applies to imported certificates from third-party vendors.
• You cannot configure the local CA when failover is enabled. You can only configure the local CA server for standalone ASASMs without failover. For more information, see CSCty43366.
• When a certificate enrollment is completed, the ASASM stores a PKCS12 file containing the user’s keypair and certificate chain, which requires about 2 KB of flash memory or disk space per enrollment. The actual amount of disk space depends on the configured RSA key size and certificate fields. Keep this guideline in mind when adding a large number of pending certificate enrollments on an ASA with a limited amount of available flash memory, because these PKCS12 files are stored in flash memory for the duration of the configured enrollment retrieval timeout.
• The lifetime ca-certificate command takes effect when the local CA server certificate is first generated (that is, when you initially configure the local CA server and issue the no shutdown command). When the CA certificate expires, the configured lifetime value is used to generate the new CA certificate. You cannot change the lifetime value for existing CA certificates.
• You should configure the ASASM to use an identity certificate to protect ASDM traffic and HTTPS traffic to the management interface. Identity certificates that are automatically generated with SCEP are regenerated after each reboot, so make sure that you manually install your own identity certificates. For an example of this procedure that applies only to SSL, see the following URL: http://www.cisco.com/en/US/products/ps6120/products_configuration_example09186a00809fcf91.shtml.
• The ASASM and the AnyConnect clients can only validate certificates in which the X520Serialnumber field (the serial number in the Subject Name) is in PrintableString format. If the serial number format uses encoding such as UTF8, the certificate authorization will fail.
Configuring Digital Certificates

This section describes how to configure local CA certificates. Make sure that you follow the sequence of tasks listed to correctly configure this type of digital certificate. This section includes the following topics:

- Configuring Key Pairs, page 38-9
- Removing Key Pairs, page 38-10
- Configuring Trustpoints, page 38-10
- Configuring CRLs for a Trustpoint, page 38-13
- Exporting a Trustpoint Configuration, page 38-15
- Importing a Trustpoint Configuration, page 38-16
- Configuring CA Certificate Map Rules, page 38-17
- Obtaining Certificates Manually, page 38-18
- Obtaining Certificates Automatically with SCEP, page 38-20
- Configuring Proxy Support for SCEP Requests, page 38-21
- Enabling the Local CA Server, page 38-22
- Configuring the Local CA Server, page 38-23
- Customizing the Local CA Server, page 38-25
- Debugging the Local CA Server, page 38-26
- Disabling the Local CA Server, page 38-26
- Deleting the Local CA Server, page 38-26
- Configuring Local CA Certificate Characteristics, page 38-27

Configuring Key Pairs

To generate key pairs, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td>crypto key generate rsa</td>
<td>Generates one, general-purpose RSA key pair. The default key modulus is 1024. To specify other modulus sizes, use the <code>modulus</code> keyword.</td>
</tr>
<tr>
<td><strong>Example:</strong> hostname/contexta(config)# crypto key generate rsa</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td>crypto key generate rsa label key-pair-label</td>
<td>(Optional) Assigns a label to each key pair. The label is referenced by the trustpoint that uses the key pair. If you do not assign a label, the key pair is automatically labeled, <code>Default-RSA-Key</code>.</td>
</tr>
<tr>
<td><strong>Example:</strong> hostname/contexta(config)# crypto key generate rsa label exchange</td>
<td></td>
</tr>
</tbody>
</table>
### Removing Key Pairs

To remove key pairs, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>crypto key zeroize rsa</td>
<td>Removes key pairs.</td>
</tr>
</tbody>
</table>

**Example:**

```
hostname(config)# crypto key zeroize rsa
```

**Examples**

The following example shows how to remove key pairs:

```
hostname(config)# crypto key zeroize rsa
WARNING: All RSA keys will be removed.
WARNING: All device certs issued using these keys will also be removed.

Do you really want to remove these keys? [yes/no] y
```

### Configuring Trustpoints

To configure a trustpoint, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>crypto ca trustpoint trustpoint-name</td>
<td>Creates a trustpoint that corresponds to the CA from which the ASASM needs to receive a certificate. Enters the crypto ca trustpoint configuration mode, which controls CA-specific trustpoint parameters that you may configure starting in Step 3.</td>
</tr>
</tbody>
</table>

**Example:**

```
hostname/contexta(config)# crypto ca trustpoint Main
```

**Step 2**

Choose one of the following options:
### Command Purpose

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>enrollment url url</td>
<td>Requests automatic enrollment using SCEP with the specified trustpoint and configures the enrollment URL.</td>
</tr>
<tr>
<td>Example: hostname/contexta(config-ca-trustpoint)# enrollment url <a href="http://10.29.67.142:80/certsrv/mscep/mscep.dll">http://10.29.67.142:80/certsrv/mscep/mscep.dll</a></td>
<td></td>
</tr>
<tr>
<td>enrollment terminal</td>
<td>Requests manual enrollment with the specified trustpoint by pasting the certificate received from the CA into the terminal.</td>
</tr>
<tr>
<td>Example: hostname/contexta(config-ca-trustpoint)# enrollment terminal</td>
<td></td>
</tr>
<tr>
<td>Step 3 revocation-check crl none</td>
<td>Specifies the available CRL configuration options.</td>
</tr>
<tr>
<td>Step 3 revocation-check crl</td>
<td>Note: To enable either required or optional CRL checking, make sure that you configure the trustpoint for CRL management after obtaining certificates.</td>
</tr>
<tr>
<td>Step 3 revocation-check none</td>
<td></td>
</tr>
<tr>
<td>Example: hostname/contexta(config-ca-trustpoint)# revocation-check crl none</td>
<td></td>
</tr>
<tr>
<td>Example: hostname/contexta(config-ca-trustpoint)# revocation-check crl</td>
<td></td>
</tr>
<tr>
<td>Example: hostname/contexta(config-ca-trustpoint)# revocation-check none</td>
<td></td>
</tr>
<tr>
<td>Step 4 crl configure</td>
<td>Enters crl configuration mode.</td>
</tr>
<tr>
<td>Example: hostname/contexta(config-ca-trustpoint)# crl configure</td>
<td></td>
</tr>
<tr>
<td>Step 5 email address</td>
<td>During enrollment, asks the CA to include the specified e-mail address in the Subject Alternative Name extension of the certificate.</td>
</tr>
<tr>
<td>Example: hostname/contexta(config-ca-trustpoint)# email example.com</td>
<td></td>
</tr>
<tr>
<td>Step 6 enrollment retry period</td>
<td>(Optional) Specifies a retry period in minutes, and applies only to SCEP enrollment.</td>
</tr>
<tr>
<td>Example: hostname/contexta(config-ca-trustpoint)# enrollment retry period 5</td>
<td></td>
</tr>
<tr>
<td>Step 7 enrollment retry count</td>
<td>(Optional) Specifies a maximum number of permitted retries, and applies only to SCEP enrollment.</td>
</tr>
<tr>
<td>Example: hostname/contexta(config-ca-trustpoint)# enrollment retry period 2</td>
<td></td>
</tr>
<tr>
<td>Step 8 fqdn fqdn</td>
<td>During enrollment, asks the CA to include the specified fully qualified domain name in the Subject Alternative Name extension of the certificate.</td>
</tr>
<tr>
<td>Example: hostname/contexta(config-ca-trustpoint)# fqdn example.com</td>
<td></td>
</tr>
</tbody>
</table>
### Configuring Digital Certificates

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td><code>ip-address ip-address</code></td>
<td>During enrollment, asks the CA to include the IP address of the ASASM in the certificate.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> &lt;br&gt;hostname/contexta(config-ca-trustpoint)# ip-address 10.10.100.1</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td><code>keypair name</code></td>
<td>Specifies the key pair whose public key is to be certified.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> &lt;br&gt;hostname/contexta(config-ca-trustpoint)# keypair exchange</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td><code>match certificate map-name override ocsp</code></td>
<td>Configures OCSP URL overrides and trustpoints to use for validating OCSP responder certificates.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> &lt;br&gt;hostname/contexta(config-ca-trustpoint)# match certificate examplemap override ocsp</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td><code>ocsp disable-nonce</code></td>
<td>Disables the nonce extension on an OCSP request. The nonce extension cryptographically binds requests with responses to avoid replay attacks.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> &lt;br&gt;hostname/contexta(config-ca-trustpoint)# ocsp disable-nonce</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td><code>ocsp url</code></td>
<td>Configures an OCSP server for the ASASM to use to check all certificates associated with a trustpoint rather than the server specified in the AIA extension of the client certificate.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> &lt;br&gt;hostname/contexta(config-ca-trustpoint)# ocsp url</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td><code>password string</code></td>
<td>Specifies a challenge phrase that is registered with the CA during enrollment. The CA usually uses this phrase to authenticate a subsequent revocation request.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> &lt;br&gt;hostname/contexta(config-ca-trustpoint)# password mypassword</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td><code>revocation check</code></td>
<td>Sets one or more methods for revocation checking: CRL, OCSP, and none.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> &lt;br&gt;hostname/contexta(config-ca-trustpoint)# revocation check</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td><code>subject-name X.500 name</code></td>
<td>During enrollment, asks the CA to include the specified subject DN in the certificate. If a DN string includes a comma, enclose the value string within double quotes (for example, O=&quot;Company, Inc.&quot;).</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> &lt;br&gt;hostname/contexta(config-ca-trustpoint)# myname X.500 examplename</td>
<td></td>
</tr>
</tbody>
</table>
Configuring Digital Certificates

Configuring CRLs for a Trustpoint

To use mandatory or optional CRL checking during certificate authentication, you must configure CRLs for each trustpoint. To configure CRLs for a trustpoint, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>crypto ca trustpoint trustpoint-name</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname (config)# crypto ca trustpoint Main</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>crl configure</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname (config-ca-trustpoint)# crl configure</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>Do one of the following:</td>
</tr>
<tr>
<td><strong>policy cdp</strong></td>
<td>Configures retrieval policy. CRLs are retrieved only from the CRL distribution points specified in authenticated certificates.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname (config-ca-crl)# policy cdp</td>
</tr>
<tr>
<td><strong>policy static</strong></td>
<td>Configures retrieval policy. CRLs are retrieved only from URLs that you configure.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname (config-ca-crl)# policy static</td>
</tr>
</tbody>
</table>

Note

SCEP retrieval is not supported by distribution points specified in certificates.

To continue, go to Step 5.

Tip

To set all CRL configuration parameters to default values, use the default command. At any time during CRL configuration, reenter this command to restart the procedure.

Note

Make sure that you have enabled CRLs before entering this command. In addition, the CRL must be available for authentication to succeed.
## Configuring Digital Certificates

### Command

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>policy both</td>
<td>Configures retrieval policy. CRLs are retrieved from CRL distribution points specified in authenticated certificates and from URLs that you configure. To continue, go to Step 4.</td>
</tr>
</tbody>
</table>

### Step 4

**url n url**

**Example:**

```
hostname (config-ca-crl)# url 2
http://www.example.com
```

If you used the keywords **static** or **both** when you configured the CRL policy, you must configure URLs for CRL retrieval. You can enter up to five URLs, ranked 1 through 5. The **n** is the rank assigned to the URL. To remove a URL, use the **no url n** command.

### Step 5

**protocol http | ldap | scep**

**Example:**

```
hostname (config-ca-crl)# protocol http
```

Configures the retrieval method. Specifies HTTP, LDAP, or SCEP as the CRL retrieval method.

### Step 6

**cache-time refresh-time**

**Example:**

```
hostname (config-ca-crl)# cache-time 420
```

Configures how long the ASASM caches CRLs for the current trustpoint. **refresh-time** is the number of minutes that the ASASM waits before considering a CRL stale.

### Step 7

Do one of the following:

- **enforcenextupdate**
  
  **Example:**
  
  ```
  hostname (config-ca-crl)# enforcenextupdate
  ```

  Requires the NextUpdate field in CRLs. This is the default setting.

- **no enforcenextupdate**
  
  **Example:**
  
  ```
  hostname (config-ca-crl)# no enforcenextupdate
  ```

  Allows the NextUpdate field to be absent in CRLs.

### Step 8

**ldap-defaults server**

**Example:**

```
hostname (config-ca-crl)# ldap-defaults ldap1
```

Identifies the LDAP server to the ASASM if LDAP is specified as the retrieval protocol. You can specify the server by DNS hostname or by IP address. You can also provide a port number if the server listens for LDAP queries on a port other than the default of 389.

**Note** If you use a hostname instead of an IP address to specify the LDAP server, make sure that you have configured the ASASM to use DNS.

### Step 9

**ldap-dn admin-DN password**

**Example:**

```
hostname (config-ca-crl)# ldap-dn
cn=admin,ou=devtest,o=engineering,c00lRunZ
```

Allows CRL retrieval if the LDAP server requires credentials.
Exporting a Trustpoint Configuration

To export a trustpoint configuration, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>crypto ca export</td>
<td>Exports a trustpoint configuration with all associated keys and</td>
</tr>
<tr>
<td>trustpoint</td>
<td>certificates in PKCS12 format. The ASASM displays the PKCS12 data</td>
</tr>
<tr>
<td>Example:</td>
<td>in the terminal. You can copy the data. The trustpoint data is password</td>
</tr>
<tr>
<td>hostname(config)#</td>
<td>protected; however, if you save the trustpoint data in a file, make</td>
</tr>
<tr>
<td>crypto ca export</td>
<td>sure that the file is in a secure location.</td>
</tr>
<tr>
<td>Main</td>
<td></td>
</tr>
</tbody>
</table>

Examples

The following example exports PKCS12 data for the trustpoint Main with the passphrase Wh0zits:

```
hostname (config)# crypto ca export Main pkcs12 Wh0zits
```

Exported pkcs12 follows:

```
[ PKCS12 data omitted ]
---End - This line not part of the pkcs12---
```
Importing a Trustpoint Configuration

To import a trustpoint configuration, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>crypto ca import trustpoint pkcs12</strong></td>
<td>Imports keypairs and issued certificates that are associated with a trustpoint configuration. The ASASM prompts you to paste the text into the terminal in base 64 format. The key pair imported with the trustpoint is assigned a label that matches the name of the trustpoint that you create.</td>
</tr>
</tbody>
</table>

**Example:**

hostname(config)# crypto ca import Main pkcs12

Enter the base 64 encoded pkcs12.
End with a blank line or the word "quit" on a line by itself:
[ PKCS12 data omitted ]
quit
INFO: Import PKCS12 operation completed successfully

**Note** If an ASASM has trustpoints that share the same CA, you can use only one of the trustpoints that share the CA to validate user certificates. To control which trustpoint that shares a CA is used for validation of user certificates issued by that CA, use the **support-user-cert-validation** keyword.

**Examples**

The following example manually imports PKCS12 data to the trustpoint Main with the passphrase Wh0zits:

hostname (config)# crypto ca import Main pkcs12 Wh0zits

Enter the base 64 encoded pkcs12.
End with a blank line or the word "quit" on a line by itself:
[ PKCS12 data omitted ]
quit
INFO: Import PKCS12 operation completed successfully

The following example manually imports a certificate for the trustpoint Main:

hostname (config)# crypto ca import Main certificate
% The fully-qualified domain name in the certificate will be:
securityappliance.example.com

Enter the base 64 encoded certificate.
End with a blank line or the word "quit" on a line by itself
[ certificate data omitted ]
quit
INFO: Certificate successfully imported
Configuring CA Certificate Map Rules

You can configure rules based on the Issuer and Subject fields of a certificate. Using the rules you create, you can map IPsec peer certificates to tunnel groups with the `tunnel-group-map` command. The ASASM supports one CA certificate map, which can include many rules.

To configure a CA certificate map rule, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Enters CA certificate map configuration mode for the rule you want to configure and specifies the rule index number.</td>
</tr>
<tr>
<td><code>crypto ca certificate map</code></td>
<td>Specifying the distinguished name of all issued certificates, which is also the subject-name DN of the self-signed CA certificate. Use commas to separate attribute-value pairs. Insert quotation marks around any value that includes a comma. An issuer-name must be less than 500 alphanumeric characters. The default issuer-name is <code>cn=hostname.domain-name</code>.</td>
</tr>
<tr>
<td><code>sequence-number</code></td>
<td>Enters CA certificate map configuration mode for the rule you want to configure and specifies the rule index number.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# crypto ca certificate map 1</td>
</tr>
</tbody>
</table>

| **Step 2**                  | Specifies tests that the ASASM can apply to values found in the Subject field of certificates. The tests can apply to specific attributes or to the entire field. You can configure many tests per rule, and all the tests you specify with these commands must be true for a rule to match a certificate. The following are valid operators: |
| `issuer-name`               | eq—The field or attribute must be identical to the value given.          |
| `DN-string`                 | ne—The field or attribute cannot be identical to the value given.        |
| Example:                    | hostname(config-ca-cert-map)# issuer-name cn=asa.example.com            |

| **Step 3**                  | Specifies tests that the ASASM can apply to values found in the Subject field of certificates. The tests can apply to specific attributes or to the entire field. You can configure many tests per rule, and all the tests you specify with these commands must be true for a rule to match a certificate. The following are valid operators: |
| `subject-name attr`         | eq—The field or attribute must be identical to the value given.          |
| `tag eq | co | ne | nc` | ne—The field or attribute cannot be identical to the value given.        |
| `string`                     | co—Part or all of the field or attribute must match the value given.     |
| Example:                    | hostname(config-ca-cert-map)# subject-name attr cn eq mycert            |

| **Step 4**                  | Saves the running configuration.                                          |
| `write memory`              | hostname (config)# write memory                                            |

Example:
hostname(config)# write memory
Obtaining Certificates Manually

To obtain certificates manually, perform the following steps:

<table>
<thead>
<tr>
<th>Step 1</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>crypto ca authenticate trustpoint</td>
<td>Imports the CA certificate for the configured trustpoint.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>hostname(config)# crypto ca authenticate Main</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Enter the base 64 encoded CA certificate.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>End with a blank line or the word &quot;quit&quot; on a line by itself</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MIIDRTCCAu+gAwIBAgIQKVcqP/KW74VPONZzL+JbRTANBgkqhkiG9w0BAQJFADCB</td>
<td></td>
</tr>
<tr>
<td></td>
<td>[ certificate data omitted ]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>/7QEM8izy0EOTSrKu7Nid76jw5e4qttkQ==</td>
<td></td>
</tr>
<tr>
<td></td>
<td>quit</td>
<td></td>
</tr>
<tr>
<td></td>
<td>INFO: Certificate has the following attributes:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Fingerprint: 24b81433 409b3fd5 e5431699 8d490d34</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Do you accept this certificate? [yes/no]: y</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Trustpoint CA certificate accepted.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>% Certificate successfully imported</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 2</th>
<th>crypto ca enroll trustpoint</th>
<th>Enrolls the ASASM with the trustpoint. Generates a certificate for signing data and depending on the type of keys that you have configured, for encrypting data.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>Example:</strong></td>
<td>Whether a trustpoint requires that you manually obtain certificates is determined by the use of the enrollment terminal command when you configure the trustpoint. For more information, see the “Configuring Trustpoints” section on page 38-10.</td>
</tr>
<tr>
<td></td>
<td>hostname(config)# crypto ca enroll Main</td>
<td></td>
</tr>
<tr>
<td></td>
<td>% Start certificate enrollment ..</td>
<td></td>
</tr>
<tr>
<td></td>
<td>% The fully-qualified domain name in the certificate will be: securityappliance.example.com</td>
<td></td>
</tr>
<tr>
<td></td>
<td>% Include the device serial number in the subject name? [yes/no]: n</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Display Certificate Request to terminal? [yes/no]: y</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Certificate Request follows:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MITBoDCCAQkCAQAwIzEHMB8GC5qGSIb3DQRJAhYRSnmYVWyXQaXgu</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Y21zY28uY29t</td>
<td></td>
</tr>
<tr>
<td></td>
<td>[ certificate data omitted ]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>jF4waw68eOxQxVmdqMWqQ+RbI0Ynvt8g6hnBTrd0OgdqjVLt</td>
<td></td>
</tr>
<tr>
<td></td>
<td>---End - This line not part of the certificate request---</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Redisplay enrollment request? [yes/no]: n</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note: This step assumes that you have already obtained a base-64 encoded CA certificate from the CA represented by the trustpoint.

Whether a trustpoint requires that you manually obtain certificates is determined by the use of the enrollment terminal command when you configure the trustpoint. For more information, see the “Configuring Trustpoints” section on page 38-10.
### Chapter 38      Configuring Digital Certificates

#### Step 3
**crypto ca import trustpoint certificate**

**Example:**
```
hostname (config)# crypto ca import Main certificate
% The fully-qualified domain name in the certificate will be: securityappliance.example.com
```

Enter the base 64 encoded certificate. End with a blank line or the word “quit” on a line by itself.
```
[ certificate data omitted ]
quit
INFO: Certificate successfully imported
```

**Purpose:** Imports each certificate you receive from the CA. Requests that you paste the certificate to the terminal in base-64 format.

#### Step 4
**show crypto ca server certificate**

**Example:**
```
hostname(config)# show crypto ca server certificate
Main
```

**Purpose:** Verifies that the enrollment process was successful by displaying certificate details issued for the ASASM and the CA certificate for the trustpoint.

#### Step 5
**write memory**

**Example:**
```
hostname(config)# write memory
```

**Purpose:** Saves the running configuration. Repeat these steps for each trustpoint that you configure for manual enrollment.

---

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 3** | **crypto ca import trustpoint certificate**<br>**Example:**<br>```
hostname (config)# crypto ca import Main certificate
% The fully-qualified domain name in the certificate will be: securityappliance.example.com
```

Enter the base 64 encoded certificate. End with a blank line or the word “quit” on a line by itself.<br>```
[ certificate data omitted ]
quit
INFO: Certificate successfully imported
```
| Imports each certificate you receive from the CA. Requests that you paste the certificate to the terminal in base-64 format. |
| **Step 4** | **show crypto ca server certificate**<br>**Example:**<br>```
hostname(config)# show crypto ca server certificate
Main
```
| Verifies that the enrollment process was successful by displaying certificate details issued for the ASASM and the CA certificate for the trustpoint. |
| **Step 5** | **write memory**<br>**Example:**<br>```
hostname(config)# write memory
```
| Saves the running configuration. Repeat these steps for each trustpoint that you configure for manual enrollment. |
## Obtaining Certificates Automatically with SCEP

To obtain certificates automatically using SCEP, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><strong>crypto ca authenticate trustpoint</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Obtain the CA certificate for the configured trustpoint.</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong> This step assumes that you have already obtained a base-64 encoded CA certificate from the CA represented by the trustpoint.</td>
</tr>
<tr>
<td></td>
<td>When you configure the trustpoint, use of the enrollment url command determines whether or not you must obtain certificates automatically via SCEP.</td>
</tr>
<tr>
<td></td>
<td>For more information, see the “Configuring Trustpoints” section on page 38-10.</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><strong>crypto ca enroll trustpoint</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Enroll the ASASM with the trustpoint. Retrieves a certificate for signing data and depending on the type of keys you have configured, for encrypting data.</td>
</tr>
<tr>
<td></td>
<td>Before entering this command, contact the CA administrator, who may need to authenticate the enrollment request manually before the CA grants certificates.</td>
</tr>
<tr>
<td></td>
<td>If the ASASM does not receive a certificate from the CA within one minute (the default) of sending a certificate request, it resends the certificate request.</td>
</tr>
<tr>
<td></td>
<td>The ASASM continues sending a certificate request each minute until a certificate is received.</td>
</tr>
<tr>
<td></td>
<td>If the fully qualified domain name configured for the trustpoint is not identical to the fully qualified domain name of the ASASM, including the case of the characters, a warning appears. To resolve this issue, exit the enrollment process, make any necessary corrections, and reenter the crypto ca enroll command.</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong> If the ASASM reboots after you have issued the crypto ca enroll command but before you have received the certificate, reenter the crypto ca enroll command and notify the CA administrator.</td>
</tr>
</tbody>
</table>
Chapter 38 Configuring Digital Certificates

Configuring Digital Certificates

Step 3
show crypto ca server certificate

Example:
hostname/contexta(config)# show crypto ca server certificate Main
Verifies that the enrollment process was successful by displaying certificate details issued for the ASASM and the CA certificate for the trustpoint.

Step 4
write memory

Example:
hostname/contexta(config)# write memory
Saves the running configuration.

Configuring Proxy Support for SCEP Requests

To configure the ASA to authenticate remote access endpoints using third-party CAs, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>crypto ikev2 enable outside client-services port portnumber</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config-tunnel-ipsec)# crypto ikev2 enable outside client-services</td>
</tr>
<tr>
<td>Command Purpose</td>
<td>Enables client services.</td>
</tr>
<tr>
<td>Note</td>
<td>Needed only if you support IKEv2.</td>
</tr>
<tr>
<td></td>
<td>Enter this command in tunnel-group ipsec-attributes configuration mode.</td>
</tr>
<tr>
<td></td>
<td>The default port number is 443.</td>
</tr>
<tr>
<td>Step 2</td>
<td>scep-enrollment enable</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config-tunnel-general)# scep-enrollment enable</td>
</tr>
<tr>
<td>INFO:</td>
<td>'authentication aaa certificate' must be configured to complete setup of this option.</td>
</tr>
<tr>
<td>Command Purpose</td>
<td>Enables SCEP enrollment for the tunnel group.</td>
</tr>
<tr>
<td></td>
<td>Enter this command in tunnel-group general-attributes configuration mode.</td>
</tr>
<tr>
<td>Step 3</td>
<td>scep-forwarding-url value URL</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config-group-policy)# scep-forwarding-url value <a href="http://ca.example.com:80/">http://ca.example.com:80/</a></td>
</tr>
<tr>
<td>Command Purpose</td>
<td>Enrolls the SCEP CA for the group policy.</td>
</tr>
<tr>
<td></td>
<td>Enter this command once per group policy to support a third-party digital certificate. Enter the command in group-policy general-attributes configuration mode.</td>
</tr>
<tr>
<td></td>
<td>URL is the SCEP URL on the CA.</td>
</tr>
<tr>
<td>Step 4</td>
<td>secondary-pre-fill-username clientless hide use-common-password password</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# tunnel-group remotegrp webvpn-attributes hostname(config-tunnel-webvpn)# secondary-pre-fill-username clientless hide use-common-password secret</td>
</tr>
<tr>
<td>Command Purpose</td>
<td>Supplies a common, secondary password when a certificate is unavailable for WebLaunch support of the SCEP proxy.</td>
</tr>
<tr>
<td></td>
<td>You must use the hide keyword to support the SCEP proxy.</td>
</tr>
<tr>
<td></td>
<td>For example, a certificate is not available to an endpoint requesting one. Once the endpoint has the certificate, AnyConnect disconnects, then reconnects to the ASA to qualify for a DAP policy that provides access to internal network resources.</td>
</tr>
</tbody>
</table>
### Enabling the Local CA Server

Before enabling the local CA server, you must first create a passphrase of at least seven characters to encode and archive a PKCS12 file that includes the local CA certificate and keypair to be generated. The passphrase unlocks the PKCS12 archive if the CA certificate or keypair is lost.

To enable the local CA server, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>crypto ca server</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname (config)# crypto ca server</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>no shutdown</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname (config-ca-server)# no shutdown</td>
</tr>
</tbody>
</table>

**Note**

After you enable the local CA server, save the configuration to make sure that the local CA certificate and keypair are not lost after a reboot occurs.

### Examples

The following example enables the local CA server:

```
hostname (config)# crypto ca server
```
hostname (config-ca-server)# no shutdown

% Some server settings cannot be changed after CA certificate generation.
% Please enter a passphrase to protect the private key
% or type Return to exit

Password: caserver

Re-enter password: caserver

Keypair generation process begin. Please wait...

The following is sample output that shows local CA server configuration and status:

Certificate Server LOCAL-CA-SERVER:
  Status: enabled
  State: enabled
  Server’s configuration is locked (enter "shutdown" to unlock it)
  Issuer name: CN=wz5520-1-16
  CA certificate fingerprint/thumbprint: (MD5)
    76dd1439 ac94fdbc 74a0a89f cb815acc
  CA certificate fingerprint/thumbprint: (SHA1)
    58754ffd 9f19f9fd b13b4b02 15b3e4be b70b5a83
  Last certificate issued serial number: 0x6
  CA certificate expiration timer: 14:25:11 UTC Jan 16 2008
  CRL NextUpdate timer: 16:09:55 UTC Jan 24 2007
  Current primary storage dir: flash:

Configuring the Local CA Server

To configure the local CA server, perform the following steps:

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>crypto ca server</td>
<td>Enters local CA server configuration mode. Generates the local CA.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>hostname (config)# crypto ca server</td>
<td></td>
</tr>
<tr>
<td>Step 2</td>
<td>smtp from-address e-mail_address</td>
<td>Specifies the SMTP from-address, a valid e-mail address that the local CA uses as a from address when sending e-mail messages that deliver OTPs for an enrollment invitation to users.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>hostname (config-ca-server)# smtp from-address <a href="mailto:SecurityAdmin@hostcorp.com">SecurityAdmin@hostcorp.com</a></td>
<td></td>
</tr>
</tbody>
</table>
### Configuring Digital Certificates

#### Step 3

**subject-name-default dn**

**Example:**

```
hostname (config-ca-server)# subject-name-default
cn=engineer, o=asc systems, c=“US”
```

(Optional) Specifies the subject-name DN that is appended to each username on issued certificates. The subject-name DN and the username combine to form the DN in all user certificates that are issued by the local CA server. If you do not specify a subject-name DN, you must specify the exact subject name DN to be included in a user certificate each time that you add a user to the user database.

**Note** Make sure that you review all optional parameters carefully before you enable the configured local CA, because you cannot change issuer-name and keysize server values after you enable the local CA for the first time.

#### Step 4

**no shutdown**

**Example:**

```
hostname (config-ca-server)# no shutdown
```

Creates the self-signed certificate and associates it with the local CA on the ASASM. The self-signed certificate key usage extension has key encryption, key signature, CRL signing, and certificate signing capabilities.

**Note** After the self-signed local CA certificate has been generated, to change any characteristics, you must delete the existing local CA server and completely recreate it.

The local CA server keeps track of user certificates, so the administrator can revoke or restore privileges as needed.

### Examples

The following example shows how to configure and enable the local CA server using the predefined default values for all required parameters:

```
hostname (config)# crypto ca server
hostname (config-ca-server) # smtp from-address SecurityAdmin@hostcorp.com
hostname (config-ca-server)# subject-name-default cn=engineer, o=asc Systems, c=US
hostname (config-ca-server)# no shutdown
```
# Customizing the Local CA Server

To configure a customized local CA server, perform the following steps:

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>crypto ca server</td>
<td>Enters local CA server configuration mode. Allows you to configure and manage a local CA.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>hostname (config)# crypto ca server</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>issuer-name DN-string</td>
<td>Specifies parameters that do not have default values.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>hostname (config-ca-server)# issuer-name cn=xx5520,cn=30.132.0.25,ou=DevTest,ou=QA,o=ASC Systems</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>smtp subject subject-line</td>
<td>Customizes the text that appears in the subject field of all e-mail messages sent from the local CA server</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>hostname (config-ca-server) # smtp subject Priority E-Mail: Enclosed Confidential Information is Required for Enrollment</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>smtp from-address e-mail_address</td>
<td>Specifies the e-mail address that is to be used as the From: field of all e-mail messages that are generated by the local CA server.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>hostname (config-ca-server) # smtp from-address <a href="mailto:SecurityAdmin@hostcorp.com">SecurityAdmin@hostcorp.com</a></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>subject-name-default dn</td>
<td>Specifies an optional subject-name DN to be appended to a username on issued certificates. The default subject-name DN becomes part of the username in all user certificates issued by the local CA server.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>hostname (config-ca-server) # subject-name default cn=engineer, o=ASC Systems, c=US</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Note</td>
<td>If you do not specify a subject-name-default to serve as a standard subject-name default, you must specify a DN each time that you add a user.</td>
</tr>
</tbody>
</table>
Debugging the Local CA Server

To debug the newly configured local CA server, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td>crypto ca server</td>
<td>Enters local ca server configuration mode. Allows you to configure and manage a local CA.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname (config)# crypto ca server</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td>debug crypto ca server</td>
<td>Displays debugging messages when you configure and enable the local CA server. Performs level 1 debugging functions; levels 1-255 are available.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname (config-ca-server)# debug crypto ca server</td>
<td></td>
</tr>
<tr>
<td><strong>Note</strong></td>
<td>Debugging commands might slow down traffic on busy networks. Levels 5 and higher are reserved for raw data dumps and should be avoided during normal debugging because of excessive output.</td>
</tr>
</tbody>
</table>

Disabling the Local CA Server

To disable the local CA server, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td>crypto ca server</td>
<td>Enters local ca server configuration mode. Allows you to configure and manage a local CA.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname (config)# crypto ca server</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td>shutdown</td>
<td>Disables the local CA server. Enables website enrollment and allows you to modify the local CA server configuration. Stores the current configuration and associated files. After initial startup, you can reenable the local CA without being prompted for the passphrase.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname (config-ca-server)# shutdown</td>
<td>INFO: Local CA Server has been shutdown.</td>
</tr>
</tbody>
</table>

Deleting the Local CA Server

To delete an existing local CA server (either enabled or disabled), enter one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Do one of the following:</td>
</tr>
</tbody>
</table>
### Configuring Digital Certificates

You can configure the following characteristics of local CA certificates:

- The name of the certificate issuer as it appears on all user certificates.
- The lifetime of the local CA certificates (server and user) and the CRL.
- The length of the public and private keypairs associated with local CA and user certificates.

This section includes the following topics:

- Configuring the Issuer Name, page 38-28
- Configuring the CA Certificate Lifetime, page 38-28
- Configuring the User Certificate Lifetime, page 38-29
- Configuring the CRL Lifetime, page 38-30
- Configuring the Server Keysize, page 38-30
- Setting Up External Local CA File Storage, page 38-31
- Downloading CRLs, page 38-33
- Storing CRLs, page 38-34
- Setting Up Enrollment Parameters, page 38-35
- Adding and Enrolling Users, page 38-36
- Renewing Users, page 38-38
- Restoring Users, page 38-39
- Removing Users, page 38-39
- Revoking Certificates, page 38-40
- Maintaining the Local CA Certificate Database, page 38-40
- Rolling Over Local CA Certificates, page 38-40
- Archiving the Local CA Server Certificate and Keypair, page 38-41

### Command Line Interface

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>no crypto ca server</td>
<td>Removes an existing local CA server (either enabled or disabled).</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname (config)# no crypto ca server</td>
</tr>
<tr>
<td>clear configure crypto ca server</td>
<td>Note: Deleting the local CA server removes the configuration from the ASASM. After the configuration has been deleted, it is unrecoverable. Make sure that you also delete the associated local CA server database and configuration files (that is, all files with the wildcard name, LOCAL-CA-SERVER.*).</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname (config)# clear config crypto ca server</td>
</tr>
</tbody>
</table>

### Configuring Local CA Certificate Characteristics

You can configure the following characteristics of local CA certificates:

- The name of the certificate issuer as it appears on all user certificates.
- The lifetime of the local CA certificates (server and user) and the CRL.
- The length of the public and private keypairs associated with local CA and user certificates.

This section includes the following topics:

- Configuring the Issuer Name, page 38-28
- Configuring the CA Certificate Lifetime, page 38-28
- Configuring the User Certificate Lifetime, page 38-29
- Configuring the CRL Lifetime, page 38-30
- Configuring the Server Keysize, page 38-30
- Setting Up External Local CA File Storage, page 38-31
- Downloading CRLs, page 38-33
- Storing CRLs, page 38-34
- Setting Up Enrollment Parameters, page 38-35
- Adding and Enrolling Users, page 38-36
- Renewing Users, page 38-38
- Restoring Users, page 38-39
- Removing Users, page 38-39
- Revoking Certificates, page 38-40
- Maintaining the Local CA Certificate Database, page 38-40
- Rolling Over Local CA Certificates, page 38-40
- Archiving the Local CA Server Certificate and Keypair, page 38-41
Configuring Digital Certificates

Configuring the Issuer Name

To configure the certificate issuer name, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>crypto ca server</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname (config)# crypto ca server</td>
</tr>
<tr>
<td>Enters local CA server configuration mode. Allows you to configure and manage a local CA.</td>
<td></td>
</tr>
</tbody>
</table>

| Step 2 | issuer-name DN-string |
| **Example:** | hostname (config-ca-server)# issuer-name CN=xx5520,CN=30.132.0.25,ou=DevTest,ou=QA,O=ABC Systems |
| Specifies the local CA certificate subject name. The configured certificate issuer name is both the subject name and issuer name of the self-signed local CA certificate, as well as the issuer name in all issued client certificates and in the issued CRL. The default issuer name in the local CA is in the format, hostname.domainname. |
| **Note** | You cannot change the issuer name value after the local CA is first enabled. |

Configuring the CA Certificate Lifetime

To configure the local CA server certificate lifetime, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>crypto ca server</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname (config)# crypto ca server</td>
</tr>
<tr>
<td>Enters local CA server configuration mode. Allows you to configure and manage a local CA.</td>
<td></td>
</tr>
</tbody>
</table>
Configuring Digital Certificates

Configuring the User Certificate Lifetime

To configure the user certificate lifetime, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>crypto ca server</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname (config)# crypto ca server</td>
</tr>
<tr>
<td></td>
<td>Enters local CA server configuration mode. Allows you to configure and manage a local CA.</td>
</tr>
<tr>
<td>Step 2</td>
<td>lifetime certificate time</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname (config-ca-server)# lifetime certificate 60</td>
</tr>
<tr>
<td></td>
<td>Sets the length of time that you want user certificates to remain valid.</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong> Before a user certificate expires, the local CA server automatically initiates certificate renewal processing by granting enrollment privileges to the user several days ahead of the certificate expiration date, setting renewal reminders, and delivering an e-mail message that includes the enrollment username and OTP for certificate renewal. Make sure that you limit the validity period of the certificate to less than the recommended end date of 03:14:08 UTC, January 19, 2038.</td>
</tr>
</tbody>
</table>
## Configuring the CRL Lifetime

To configure the CRL lifetime, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>crypto ca server</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname (config)# crypto ca server</td>
</tr>
<tr>
<td>Enters local CA server configuration mode. Allows you to configure and manage a local CA.</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>lifetime crl time</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname (config-ca-server)# lifetime crl 10</td>
</tr>
<tr>
<td>Sets the length of time that you want the CRL to remain valid. The local CA updates and reissues the CRL each time that a user certificate is revoked or unrevoked, but if no revocation changes occur, the CRL is reissued automatically once each CRL lifetime. If you do not specify a CRL lifetime, the default time period is six hours.</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>crypto ca server crl issue</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname(config)# crypto ca server crl issue</td>
</tr>
<tr>
<td>A new CRL has been issued.</td>
<td></td>
</tr>
<tr>
<td>Forces the issuance of a CRL at any time, which immediately updates and regenerates a current CRL to overwrite the existing CRL.</td>
<td></td>
</tr>
</tbody>
</table>

### Note
Do not use this command unless the CRL file has been removed in error or has been corrupted and must be regenerated.

## Configuring the Server Keysize

To configure the server keysize, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>crypto ca server</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname (config)# crypto ca server</td>
</tr>
<tr>
<td>Enters local CA server configuration mode. Allows you to configure and manage a local CA.</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>keysize server</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname (config-ca-server)# keysize server 2048</td>
</tr>
<tr>
<td>Specifies the size of the public and private keys generated at user-certificate enrollment. The keypair size options are 512, 768, 1024, 2048 bits, and the default value is 1024 bits.</td>
<td></td>
</tr>
<tr>
<td><strong>Note</strong></td>
<td>After you have enabled the local CA, you cannot change the local CA keysize, because all issued certificates would be invalidated. To change the local CA keysize, you must delete the current local CA and reconfigure a new one.</td>
</tr>
</tbody>
</table>
Examples

The following is sample output that shows two user certificates in the database.

Username: emily1
Renewal allowed until: Not Allowed
Number of times user notified: 0
PKCS12 file stored until: 12:45:52 UTC Fri Jan 4 2017
Certificates Issued:
serial: 0x71
issued: 12:45:52 UTC Thu Jan 3 2008
expired: 12:17:37 UTC Sun Dec 31 2017
status: Not Revoked
Username: fred1
Renewal allowed until: Not Allowed
Number of times user notified: 0
PKCS12 file stored until: 12:27:59 UTC Fri Jan 4 2008
Certificates Issued:
serial: 0x2
issued: 12:27:59 UTC Thu Jan 3 2008
expired: 12:17:37 UTC Sun Dec 31 2017
status: Not Revoked

Setting Up External Local CA File Storage

You can store the local CA server configuration, users, issued certificates, and CRLs in the local CA server database either in flash memory or in an external local CA file system. To configure external local CA file storage, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><strong>mount name type</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname (config)# mount mydata type cifs</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><strong>mount name type cifs</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname (config-mount-cifs)# mount mydata type cifs server 99.1.1.99 share myshare domain example.com username user6 password ******** status enable</td>
</tr>
</tbody>
</table>

Note: Only the user who mounts a file system can unmount it with the no mount command.
### Configuring Digital Certificates

#### Examples

The following example shows the list of local CA files that appear in flash memory or in external storage:

```command
hostname (config-ca-server)# dir LOCAL* //
Directory of disk0:/LOCAL*  
...
75  -rwx 32 13:07:49 Jan 20 2007 LOCAL-CA-SERVER.ser
77  -rwx 229 13:07:49 Jan 20 2007 LOCAL-CA-SERVER.cdb
69  -rwx 0 01:09:28 Jan 20 2007 LOCAL-CA-SERVER.udb
81  -rwx 232 19:09:10 Jan 20 2007 LOCAL-CA-SERVER.crl
72  -rwx 1603 01:09:28 Jan 20 2007 LOCAL-CA-SERVER.pem
```

127119360 bytes total (79693824 bytes free)

### Command Purpose

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| Step 3  | crypto ca server
Example: hostname (config)# crypto ca server |
| Enters local CA server configuration mode. Allows you to configure and manage a local CA. |
| Step 4  | database path mount-name directory-path
Example: hostname (config-ca-server)# database path mydata:newuser |
| Specifies the location of mydata, the premounted CIFS file system to be used for the local CA server database. Establishes a path to the server and then specifies the local CA file or folder name to use for storage and retrieval. To return local CA file storage to the ASASM flash memory, use the no database path command. |
| Note To secure stored local CA files on an external server requires a premounted file system of file type CIFS or FTP that is username-protected and password-protected. |
| Step 5  | write memory
Example: hostname (config)# write memory |
| Saves the running configuration. |
| For external local CA file storage, each time that you save the ASASM configuration, user information is saved from the ASASM to the premounted file system and file location, mydata:newuser. |
| For flash memory storage, user information is saved automatically to the default location for the start-up configuration. |
## Downloading CRLs

To make the CRL available for HTTP download on a given interface or port, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><strong>crypto ca server</strong>&lt;br&gt;Example: hostname (config)# crypto ca server</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><strong>publish-crl interface interface port portnumber</strong>&lt;br&gt;Example: hostname (config-ca-server)# publish-crl outside 70</td>
</tr>
</tbody>
</table>
## Storing CRLs

To establish a specific location for the automatically generated CRL of the local CA, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> crypto ca server</td>
<td>Enters local ca server configuration mode. Allows you to configure and manage a local CA.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname (config)# crypto ca server</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> cdp-url url</td>
<td>Specifies the CDP to be included in all issued certificates. If you do not configure a specific location for the CDP, the default URL location is <a href="http://hostname.domain/CSCOCA+/asa_ca.crl">http://hostname.domain/CSCOCA+/asa_ca.crl</a>. The local CA updates and reissues the CRL each time a user certificate is revoked or unrevoked. If no revocation changes occur, the CRL is reissued once each CRL lifetime. If this command is set to serve the CRL directly from the local CA ASASM, see the “Downloading CRLs” section on page 38-33 for instructions about opening a port on an interface to make the CRL accessible from that interface. The CRL exists for other devices to validate the revocation of certificates issued by the local CA. In addition, the local CA tracks all issued certificates and status within its own certificate database. Revocation checking is performed when a validating party needs to validate a user certificate by retrieving the revocation status from an external server, which might be the CA that issued the certificate or a server designated by the CA.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config-ca-server)# cdp-url <a href="http://172.16.1.1/pathname/myca.crl">http://172.16.1.1/pathname/myca.crl</a></td>
<td></td>
</tr>
</tbody>
</table>
### Setting Up Enrollment Parameters

To set up enrollment parameters, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> crypto ca server</td>
<td>Enters local ca server configuration mode. Allows you to configure and manage a local CA.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname (config)# crypto ca server</td>
</tr>
<tr>
<td><strong>Step 2</strong> otp expiration timeout</td>
<td>Specifies the number of hours that an issued OTP for the local CA enrollment page is valid. The default expiration time is 72 hours.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config-ca-server)# otp expiration 24</td>
</tr>
<tr>
<td><strong>Step 3</strong> enrollment-retrieval timeout</td>
<td>Specifies the number of hours an already-enrolled user can retrieve a PKCS12 enrollment file. This time period begins when the user is successfully enrolled. The default retrieval period is 24 hours. Valid values for the retrieval period range from 1 to 720 hours. The enrollment retrieval period is independent of the OTP expiration period. After the enrollment retrieval time expires, the user certificate and keypair are no longer available. The only way a user may receive a certificate is for the administrator to reinitialize certificate enrollment and allow a user to log in again.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config-ca-server)# enrollment-retrieval 120</td>
</tr>
</tbody>
</table>
Adding and Enrolling Users

To add a user who is eligible for enrollment in the local CA database, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>crypto ca server user-db add username [dn dn] [email emailaddress]</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname (config-ca-server)# crypto ca server user-db add user1 dn <a href="mailto:user1@example.com">user1@example.com</a>, Engineer, Example Company, US, email <a href="mailto:user1@example.com">user1@example.com</a></td>
</tr>
<tr>
<td></td>
<td>Adds a new user to the local CA database. Options are as follows:</td>
</tr>
<tr>
<td></td>
<td>• <em>username</em>—A string of 4-64 characters, which is the simple username for the user being added. The username can be an e-mail address, which then is used to contact the user as necessary for enrollment invitations.</td>
</tr>
<tr>
<td></td>
<td>• <em>dn</em>—The distinguished name, a global, authoritative name of an entry in the OSI Directory (X.500) (for example, cn=<a href="mailto:user1@example.com">user1@example.com</a>, cn=Engineer, o=Example Company, c=US).</td>
</tr>
<tr>
<td></td>
<td>• <em>e-mail-address</em>—The e-mail address of the new user to which OTPs and notices are to be sent.</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>crypto ca server user-db allow user</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname (config-ca-server)# crypto ca server user-db allow user6</td>
</tr>
<tr>
<td></td>
<td>Provides user privileges to a newly added user.</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>crypto ca server user-db email-otp username</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname (config-ca-server)# crypto ca server user-db email-otp exampleuser1</td>
</tr>
<tr>
<td></td>
<td>Notifies a user in the local CA database to enroll and download a user certificate, which automatically e-mails the OTP to that user.</td>
</tr>
<tr>
<td><strong>Note</strong></td>
<td>When an administrator wants to notify a user through e-mail, the administrator must specify the e-mail address in the username field or in the e-mail field when adding that user.</td>
</tr>
</tbody>
</table>
### Chapter 38  Configuring Digital Certificates

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td><code>crypto ca server user-db show-otp</code></td>
<td>Shows the issued OTP.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong>&lt;br&gt;hostname (config-ca-server)# crypto ca server user-db show-otp</td>
<td></td>
</tr>
</tbody>
</table>
| 5    | `otp expiration timeout` | Sets the enrollment time limit in hours. The default expiration time is 72 hours. The `otp expiration` command defines the amount of time that the OTP is valid for user enrollment. This time period begins when the user is allowed to enroll.

After a user enrolls successfully within the time limit and with the correct OTP, the local CA server creates a PKCS12 file, which includes a keypair for the user and a user certificate that is based on the public key from the keypair generated and the subject-name DN specified when the user is added. The PKCS12 file contents are protected by a passphrase, the OTP. The OTP can be handled manually, or the local CA can e-mail this file to the user to download after the administrator allows enrollment.

The PKCS12 file is saved to temporary storage with the name, `username.p12`. With the PKCS12 file in storage, the user can return within the enrollment-retrieval time period to download the PKCS12 file as many times as needed. When the time period expires, the PKCS12 file is removed from storage automatically and is no longer available to download.

**Note** If the enrollment period expires before the user retrieves the PKCS12 file that includes the user certificate, enrollment is not permitted. |
## Renewing Users

To specify the timing of renewal notices, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> crypto ca server</td>
<td>Enters local CA server configuration mode. Allows you to configure and manage a local CA.</td>
</tr>
<tr>
<td>Example: hostname (config)# crypto ca server</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> renewal-reminder time</td>
<td>Specifies the number of days (1-90) before the local CA certificate expires that an initial reminder to reenroll is sent to certificate owners. If a certificate expires, it becomes invalid.</td>
</tr>
<tr>
<td>Example: hostname (config-ca-server)# renewal-reminder 7</td>
<td>Renewal notices and the times they are e-mailed to users are variable, and can be configured by the administrator during local CA server configuration. Three reminders are sent. An e-mail is automatically sent to the certificate owner for each of the three reminders, provided an e-mail address is specified in the user database. If no e-mail address exists for the user, a syslog message alerts you of the renewal requirement. The ASASM automatically grants certificate renewal privileges to any user who holds a valid certificate that is about to expire, as long as the user still exists in the user database. Therefore, if an administrator does not want to allow a user to renew automatically, the administrator must remove the user from the database before the renewal time period.</td>
</tr>
</tbody>
</table>
Restoring Users

To restore a user and a previously revoked certificate that was issued by the local CA server, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td><code>crypto ca server</code> Enters local ca server configuration mode. Allows you to configure and manage a local CA.</td>
</tr>
<tr>
<td>Example:</td>
<td><code>hostname (config)# crypto ca server</code></td>
</tr>
<tr>
<td>Step 2</td>
<td><code>crypto ca server unrevoke cert-serial-no</code> Restores a user and unrevokes a previously revoked certificate that was issued by the local CA server. The local CA maintains a current CRL with serial numbers of all revoked user certificates. This list is available to external devices and can be retrieved directly from the local CA if it is configured to do so with the <code>cdp-url</code> command and the <code>publish-crl</code> command. When you revoke (or unrevoke) any current certificate by certificate serial number, the CRL automatically reflects these changes.</td>
</tr>
<tr>
<td>Example:</td>
<td><code>hostname (config)# crypto ca server unrevoke 782ea09f</code></td>
</tr>
</tbody>
</table>

Removing Users

To delete a user from the user database by username, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td><code>crypto ca server</code> Enters local ca server configuration mode. Allows you to configure and manage a local CA.</td>
</tr>
<tr>
<td>Example:</td>
<td><code>hostname (config)# crypto ca server</code></td>
</tr>
<tr>
<td>Step 2</td>
<td><code>crypto ca server user-db remove username</code> Removes a user from the user database and allows revocation of any valid certificates that were issued to that user.</td>
</tr>
<tr>
<td>Example:</td>
<td><code>hostname (config)# crypto ca server user-db remove user1</code></td>
</tr>
</tbody>
</table>
Revoking Certificates

To revoke a user certificate, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>crypto ca server</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname (config)# crypto ca server</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>crypto ca server revoke cert-serial-no</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname (config-ca-server)# crypto ca server revoke 782ea09f</td>
</tr>
</tbody>
</table>

Note: The password is also required if the certificate for the ASASM needs to be revoked, so make sure that you record it and store it in a safe place.

Maintaining the Local CA Certificate Database

To maintain the local CA certificate database, make sure that you save the certificate database file, LOCAL-CA-SERVER.cdb, with the **write memory** command each time that a change to the database occurs. The local CA certificate database includes the following files:

- The LOCAL-CA-SERVER.p12 file is the archive of the local CA certificate and keypair that is generated when the local CA server is initially enabled.
- The LOCAL-CA-SERVER.crl file is the actual CRL.
- The LOCAL-CA-SERVER.ser file keeps track of the issued certificate serial numbers.

Rolling Over Local CA Certificates

Thirty days before the local CA certificate expires, a rollover replacement certificate is generated, and a syslog message informs the administrator that it is time for local CA rollover. The new local CA certificate must be imported onto all necessary devices before the current certificate expires. If the administrator does not respond by installing the rollover certificate as the new local CA certificate, validations may fail.

The local CA certificate rolls over automatically after expiration using the same keypair. The rollover certificate is available for export in base 64 format.

Examples

The following example shows a base 64 encoded local CA certificate:

MIIXlwIBAzCCF1EBCGSSqSi1b3DQEHAAcCCFC0IEghe+MIIX0jCCFzYGCSSqSi1b3DQEHBBqCCFycwghejAgEAMIXHAYJKoZIhvCNCQcBMBsGClqGSSi1b3DQEMAQwMDQ4Qjph4SxJyYtcAQGAggqhw3v4bFy+GGG2dJnB4OlPheSN+IG3SDOjDwZG9nI5vmIeoxd7Hcknxbum066DrjWxNz1Igw+GDS14q1NEi1GeP2YrJ94/NQ2z+4k+u2Zzc3R1Ke7EisE4L0fS3CuwTjx2NUNYyWmocsp4icCIELj3h7VXy6gbq2AC8l+g57+QG5vgsI5H15imwtyfUwPMDQxawZPzroG1sBqdpPa1jBHGZuZsmXm3jJ2dQ3Atro9GnIsRHyV39fcBwz4fEabH97/Vanb+fj81d5n10lJjDYy8P86tvbZ2yOVR6aKFVIB0b3Ac6PwofC9U8Z/aF3BCyMN2s2XrXva94CaYrzyotzdAkSYAKWScyEcGdquBeGDKOnctknfgy0XM+fG5rb3qAYy1GjkFQ58m9Do66RUR0oeG1DSrQKeg/hj...
Archiving the Local CA Server Certificate and Keypair

To archive the local CA server certificate and keypair, enter the following command:

```
hostname# copy LOCAL-CA-SERVER_0001.pl2
tftp://10.1.1.22/user6/
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>copy</td>
<td>Copies the local CA server certificate and keypair and all files from</td>
</tr>
<tr>
<td></td>
<td>the ASASM using either FTP or TFTP.</td>
</tr>
<tr>
<td></td>
<td>Note Make sure that you back up all local CA files as often as possible.</td>
</tr>
</tbody>
</table>

Monitoring Digital Certificates

To display certificate configuration and database information, enter one or more of the following commands:

```
hostname# show crypto ca server
hostname# show crypto ca server cert-db
hostname# show crypto ca server certificate
hostname# show crypto ca server crl
hostname# show crypto ca server user-db allowed
hostname# show crypto ca server user-db enrolled
hostname# show crypto ca server user-db expired
hostname# show crypto ca server user-db on-hold
hostname# show crypto key name of key
hostname# show running-config
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show crypto ca server</td>
<td>Shows local CA configuration and status.</td>
</tr>
<tr>
<td>show crypto ca server cert-db</td>
<td>Shows user certificates issued by the local CA.</td>
</tr>
<tr>
<td>show crypto ca server certificate</td>
<td>Shows local CA certificates on the console in base 64 format and the rollover certificate when available, including the rollover certificate thumbprint for verification of the new certificate during import onto other devices.</td>
</tr>
<tr>
<td>show crypto ca server crl</td>
<td>Shows CRLs.</td>
</tr>
<tr>
<td>show crypto ca server user-db</td>
<td>Shows users and their status, which can be used with the following qualifications to reduce the number of displayed records:</td>
</tr>
<tr>
<td></td>
<td>• allowed. Shows only users currently allowed to enroll.</td>
</tr>
<tr>
<td></td>
<td>• enrolled. Shows only users that are enrolled and hold a valid certificate</td>
</tr>
<tr>
<td></td>
<td>• expired. Shows only users holding expired certificates.</td>
</tr>
<tr>
<td></td>
<td>• on-hold. Lists only users without a certificate and not currently allowed to enroll.</td>
</tr>
<tr>
<td>show crypto ca server user-db allowed</td>
<td>Shows users who are eligible to enroll.</td>
</tr>
<tr>
<td>show crypto ca server user-db enrolled</td>
<td>Shows enrolled users with valid certificates.</td>
</tr>
<tr>
<td>show crypto ca server user-db expired</td>
<td>Shows users with expired certificates.</td>
</tr>
<tr>
<td>show crypto ca server user-db on-hold</td>
<td>Shows users without certificates who are not allowed to enroll.</td>
</tr>
<tr>
<td>show crypto key name of key</td>
<td>Shows key pairs that you have generated.</td>
</tr>
<tr>
<td>show running-config</td>
<td>Shows local CA certificate map rules.</td>
</tr>
</tbody>
</table>
Examples

The following example shows an RSA general-purpose key:

```
hostname/contexta(config)# show crypto key mypubkey
Key pair was generated at: 16:39:47 central Feb 10 2010
Key name: <Default-RSA-Key>
Usage: General Purpose Key
Modulus Size (bits): 1024
Key Data:
30819f30 0d06092a 864886f7 0d010101 05000381 8d003081 89028181 00ea51b7
0781848f 78bccac2 4a1b5b8d 2f3e30b4 4cae9f86 f4485207 159108c9 f5e49103
9eeb0f5d 45fd1811 3b4aafce 292b3b64 b4124a6f 7a777b08 75b88df1 8092a9f8
5508e9e6 2c271f24 7fd1c0c3 3aa1f1e04 c7c4efda 600f4c4a 6afe56ad c1d2c01c
e08407dd 45d9e36e 8cc0bffd 14f9e6ac eca141e4 276d7358 f7f50d13 79020301 0001
Key pair was generated at: 16:34:54 central Feb 10 2010
```

The following example shows the local CA CRL:

```
hostname (config)# show crypto ca server crl
Certificate Revocation List:
  Issuer: cn=xx5520-1-3-2007-1
  This Update: 13:32:53 UTC Jan 4 2010
  Next Update: 13:32:53 UTC Feb 3 2010
  Number of CRL entries: 2
  CRL size: 270 bytes
Revoked Certificates:
  Serial Number: 0x6f
  Revocation Date: 12:30:01 UTC Jan 4 2010
  Serial Number: 0x47
  Revocation Date: 13:32:48 UTC Jan 4 2010
```

The following example shows one user on-hold:

```
hostname (config)# show crypto ca server user-db on-hold
username: wilma101
email: <None>
dn: <None>
allowed: <not allowed>
notified: 0
hostname (config)#
```

The following example shows output of the `show running-config` command, in which local CA certificate map rules appear:

```
crypto ca certificate map 1
  issuer-name co asc
  subject-name attr ou eq Engineering
```
# Feature History for Certificate Management

Table 38-1 lists each feature change and the platform release in which it was implemented.

## Table 38-1 Feature History for Certificate Management

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Certificate management</td>
<td>7.0(1)</td>
<td>Digital certificates (including CA certificates, identity certificates, and code signer certificates) provide digital identification for authentication. A digital certificate includes information that identifies a device or user, such as the name, serial number, company, department, or IP address. CAs are trusted authorities that “sign” certificates to verify their authenticity, thereby guaranteeing the identity of the device or user. CAs issue digital certificates in the context of a PKI, which uses public-key or private-key encryption to ensure security.</td>
</tr>
</tbody>
</table>
| Certificate management        | 7.2(1)            | We introduced the following commands: 
issuer-name DN-string, revocation-check crl none, revocation-check crl, revocation-check none  
We deprecated the following commands: crl \{required | optional | nocheck\}. |

---
### Table 38-1 Feature History for Certificate Management (continued)

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Certificate management</td>
<td>8.0(2)</td>
<td>We introduced the following commands:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>cdp-url, crypto ca server, crypto ca server crl issue, crypto ca server revoke cert-serial-no, crypto ca server unrevoke cert-serial-no, crypto ca server user-db add user [dn dn] [email e-mail-address], crypto ca server user-db allow {username</td>
</tr>
<tr>
<td>SCEP proxy</td>
<td>8.4(1)</td>
<td>We introduced this feature, which provides secure deployment of device certificates from third-party CAs.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>We introduced the following commands:</td>
</tr>
</tbody>
</table>
PART 10

Configuring Application Inspection
CHAPTER 39

Getting Started with Application Layer Protocol Inspection

This chapter describes how to configure application layer protocol inspection. Inspection engines are required for services that embed IP addressing information in the user data packet or that open secondary channels on dynamically assigned ports. These protocols require the ASASM to do a deep packet inspection instead of passing the packet through the fast path (see the “Stateful Inspection Overview” section on page 1-10 for more information about the fast path). As a result, inspection engines can affect overall throughput. Several common inspection engines are enabled on the ASASM by default, but you might need to enable others depending on your network.

This chapter includes the following sections:

- Information about Application Layer Protocol Inspection, page 39-1
- Guidelines and Limitations, page 39-3
- Default Settings, page 39-4
- Configuring Application Layer Protocol Inspection, page 39-6

Information about Application Layer Protocol Inspection

This section includes the following topics:

- How Inspection Engines Work, page 39-1
- When to Use Application Protocol Inspection, page 39-2

How Inspection Engines Work

As illustrated in Figure 39-1, the ASASM uses three databases for its basic operation:

- Access lists—Used for authentication and authorization of connections based on specific networks, hosts, and services (TCP/UDP port numbers).
- Inspections—Contains a static, predefined set of application-level inspection functions.
- Connections (XLATE and CONN tables)—Maintains state and other information about each established connection. This information is used by the Adaptive Security Algorithm and cut-through proxy to efficiently forward traffic within established sessions.
Information about Application Layer Protocol Inspection

Chapter 39      Getting Started with Application Layer Protocol Inspection

Figure 39-1 How Inspection Engines Work

In Figure 39-1, operations are numbered in the order they occur, and are described as follows:

1. A TCP SYN packet arrives at the ASASM to establish a new connection.
2. The ASASM checks the access list database to determine if the connection is permitted.
3. The ASASM creates a new entry in the connection database (XLATE and CONN tables).
4. The ASASM checks the Inspections database to determine if the connection requires application-level inspection.
5. After the application inspection engine completes any required operations for the packet, the ASASM forwards the packet to the destination system.
6. The destination system responds to the initial request.
7. The ASASM receives the reply packet, looks up the connection in the connection database, and forwards the packet because it belongs to an established session.

The default configuration of the ASASM includes a set of application inspection entries that associate supported protocols with specific TCP or UDP port numbers and that identify any special handling required.

When to Use Application Protocol Inspection

When a user establishes a connection, the ASASM checks the packet against access lists, creates an address translation, and creates an entry for the session in the fast path, so that further packets can bypass time-consuming checks. However, the fast path relies on predictable port numbers and does not perform address translations inside a packet.

Many protocols open secondary TCP or UDP ports. The initial session on a well-known port is used to negotiate dynamically assigned port numbers.

Other applications embed an IP address in the packet that needs to match the source address that is normally translated when it goes through the ASASM.

If you use applications like these, then you need to enable application inspection.
When you enable application inspection for a service that embeds IP addresses, the ASASM translates embedded addresses and updates any checksum or other fields that are affected by the translation.

When you enable application inspection for a service that uses dynamically assigned ports, the ASASM monitors sessions to identify the dynamic port assignments, and permits data exchange on these ports for the duration of the specific session.

**Guidelines and Limitations**

This section includes the guidelines and limitations for this feature.

**Context Mode Guidelines**
Supported in single and multiple context mode.

**Firewall Mode Guidelines**
Supported in routed and transparent firewall mode.

**Failover Guidelines**
State information for multimedia sessions that require inspection are not passed over the state link for stateful failover. The exception is GTP, which is replicated over the state link.

**IPv6 Guidelines**
Supports IPv6 for the following inspections:
- FTP
- HTTP
- ICMP
- SIP
- SMTP
- IPsec pass-through

**Additional Guidelines and Limitations**
Some inspection engines do not support PAT, NAT, outside NAT, or NAT between same security interfaces. See “Default Settings” for more information about NAT support.

For all the application inspections, the adaptive security appliance limits the number of simultaneous, active data connections to 200 connections. For example, if an FTP client opens multiple secondary connections, the FTP inspection engine allows only 200 active connections and the 201 connection is dropped and the adaptive security appliance generates a system error message.

Inspected protocols are subject to advanced TCP-state tracking, and the TCP state of these connections is not automatically replicated. While these connections are replicated to the standby unit, there is a best-effort attempt to re-establish a TCP state.

**Inspection Reset Behavior**
When you configure an inspection engine to use a reset action and a packet triggers a reset, the ASA sends a TCP reset under the following conditions:
- The ASA sends a TCP reset to the inside host when the `service resetoutbound` command is enabled. (The `service resetoutbound` command is disabled by default.)
• The ASA sends a TCP reset to the outside host when the service resetinbound command is enabled. (The service resetinbound command is disabled by default.)

For more information, see the service command in the ASA command reference.

This behavior ensures that a reset action will reset the connections on the ASA and on inside servers; therefore countering denial of service attacks. For outside hosts, the ASA does not send a reset by default and information is not revealed through a TCP reset.

Default Settings

By default, the configuration includes a policy that matches all default application inspection traffic and applies inspection to the traffic on all interfaces (a global policy). Default application inspection traffic includes traffic to the default ports for each protocol. You can only apply one global policy, so if you want to alter the global policy, for example, to apply inspection to non-standard ports, or to add inspections that are not enabled by default, you need to either edit the default policy or disable it and apply a new one.

Table 39-1 lists all inspections supported, the default ports used in the default class map, and the inspection engines that are on by default, shown in bold. This table also notes any NAT limitations.

<table>
<thead>
<tr>
<th>Application</th>
<th>Default Port</th>
<th>NAT Limitations</th>
<th>Standards</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTIQBE</td>
<td>TCP/2748</td>
<td>No extended PAT.</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>DCERPC</td>
<td>TCP/135</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>DNS over UDP</td>
<td>UDP/53</td>
<td>No NAT support is available for name resolution through WINS.</td>
<td>RFC 1123</td>
<td>No PTR records are changed.</td>
</tr>
<tr>
<td>FTP</td>
<td>TCP/21</td>
<td>—</td>
<td>RFC 959</td>
<td>—</td>
</tr>
<tr>
<td>GTP</td>
<td>UDP/3386</td>
<td>No extended PAT.</td>
<td>—</td>
<td>Requires a special license.</td>
</tr>
<tr>
<td>H.323 H.225 and RAS</td>
<td>TCP/1720, UDP/1718, UDP (RAS) 1718-1719</td>
<td>No NAT on same security interfaces. No static PAT. No extended PAT.</td>
<td>ITU-T H.323, H.245, H225.0, Q.931, Q.932</td>
<td>—</td>
</tr>
<tr>
<td>HTTP</td>
<td>TCP/80</td>
<td>—</td>
<td>RFC 2616</td>
<td>Beware of MTU limitations stripping ActiveX and Java. If the MTU is too small to allow the Java or ActiveX tag to be included in one packet, stripping may not occur.</td>
</tr>
<tr>
<td>ICMP</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>All ICMP traffic is matched in the default class map.</td>
</tr>
<tr>
<td>ICMP ERROR</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>All ICMP traffic is matched in the default class map.</td>
</tr>
<tr>
<td>ILS (LDAP)</td>
<td>TCP/389</td>
<td>No extended PAT.</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Instant Messaging (IM)</td>
<td>Varies by client</td>
<td>No extended PAT.</td>
<td>RFC 3860</td>
<td>—</td>
</tr>
</tbody>
</table>
The default policy configuration includes the following commands:

**Table 39-1  Supported Application Inspection Engines (continued)**

<table>
<thead>
<tr>
<th>Application</th>
<th>Default Port</th>
<th>NAT Limitations</th>
<th>Standards^1</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP Options</td>
<td>—</td>
<td>—</td>
<td>RFC 791, RFC 2113</td>
<td>All IP Options traffic is matched in the default class map.</td>
</tr>
<tr>
<td>MGCP</td>
<td>UDP/2427, 2727</td>
<td>No extended PAT.</td>
<td>RFC 2705bis-05</td>
<td>—</td>
</tr>
<tr>
<td>MMP</td>
<td>TCP 5443</td>
<td>No extended PAT.</td>
<td>—</td>
<td></td>
</tr>
<tr>
<td>NetBIOS Name Server over IP</td>
<td>UDP/137, 138 (Source ports)</td>
<td>No extended PAT.</td>
<td>—</td>
<td>NetBIOS is supported by performing NAT of the packets for NBNS UDP port 137 and NBDS UDP port 138.</td>
</tr>
<tr>
<td>PPTP</td>
<td>TCP/1723</td>
<td>—</td>
<td>RFC 2637</td>
<td></td>
</tr>
<tr>
<td>RADIUS Accounting</td>
<td>1646</td>
<td>—</td>
<td>RFC 2865</td>
<td></td>
</tr>
<tr>
<td>RSH</td>
<td>TCP/514</td>
<td>No PAT</td>
<td>Berkeley UNIX</td>
<td></td>
</tr>
<tr>
<td>RTSP</td>
<td>TCP/554</td>
<td>No extended PAT.</td>
<td>RFC 2326, 2327, 1889</td>
<td>No handling for HTTP cloaking.</td>
</tr>
<tr>
<td>SIP</td>
<td>TCP/5060, UDP/5060</td>
<td>No outside NAT.</td>
<td>RFC 2543</td>
<td></td>
</tr>
<tr>
<td>SKINNY (SCCP)</td>
<td>TCP/2000</td>
<td>No outside NAT.</td>
<td>—</td>
<td>Does not handle TFTP uploaded Cisco IP Phone configurations under certain circumstances.</td>
</tr>
<tr>
<td>SMTP and ESMTP</td>
<td>TCP/25</td>
<td>—</td>
<td>RFC 821, 1123</td>
<td></td>
</tr>
<tr>
<td>SNMP</td>
<td>UDP/161, 162</td>
<td>No NAT or PAT.</td>
<td>RFC 1155, 1157, 1212, 1213, 1215</td>
<td>v.2 RFC 1902-1908; v.3 RFC 2570-2580.</td>
</tr>
<tr>
<td>SQL*Net</td>
<td>TCP/1521</td>
<td>No extended PAT.</td>
<td>—</td>
<td>v.1 and v.2.</td>
</tr>
<tr>
<td>Sun RPC over UDP and TCP</td>
<td>UDP/111</td>
<td>No extended PAT.</td>
<td>—</td>
<td>The default rule includes UDP port 111; if you want to enable Sun RPC inspection for TCP port 111, you need to create a new rule that matches TCP port 111 and performs Sun RPC inspection.</td>
</tr>
<tr>
<td>TFTP</td>
<td>UDP/69</td>
<td>—</td>
<td>RFC 1350</td>
<td>Payload IP addresses are not translated.</td>
</tr>
<tr>
<td>WAAS</td>
<td>—</td>
<td>No extended PAT.</td>
<td>—</td>
<td></td>
</tr>
<tr>
<td>XDCMP</td>
<td>UDP/177</td>
<td>No extended PAT.</td>
<td>—</td>
<td></td>
</tr>
</tbody>
</table>

1. Inspection engines that are enabled by default for the default port are in bold.

2. The ASASM is in compliance with these standards, but it does not enforce compliance on packets being inspected. For example, FTP commands are supposed to be in a particular order, but the ASASM does not enforce the order.

The default policy configuration includes the following commands:
Configuring Application Layer Protocol Inspection

This feature uses Modular Policy Framework to create a service policy. Service policies provide a consistent and flexible way to configure ASASM features. For example, you can use a service policy to create a timeout configuration that is specific to a particular TCP application, as opposed to one that applies to all TCP applications. See Chapter 30, “Configuring a Service Policy Using the Modular Policy Framework,” for more information. For some applications, you can perform special actions when you enable inspection. See Chapter 30, “Configuring a Service Policy Using the Modular Policy Framework,” for more information.

Inspection is enabled by default for some applications. See the “Default Settings” section for more information. Use this section to modify your inspection policy.

Detailed Steps

Step 1
To identify the traffic to which you want to apply inspections, add either a Layer 3/4 class map for through traffic or a Layer 3/4 class map for management traffic. See the “Creating a Layer 3/4 Class Map for Through Traffic” section on page 30-12 and “Creating a Layer 3/4 Class Map for Management Traffic” section on page 30-14 for detailed information. The management Layer 3/4 class map can be used only with the RADIUS accounting inspection.

The default Layer 3/4 class map for through traffic is called “inspection_default.” It matches traffic using a special match command, match default-inspection-traffic, to match the default ports for each application protocol. This traffic class (along with match any, which is not typically used for inspection) matches both IPv4 and IPv6 traffic for inspections that support IPv6. See the “Guidelines and Limitations” section on page 39-3 for a list of IPv6-enabled inspections.

You can specify a match access-list command along with the match default-inspection-traffic command to narrow the matched traffic to specific IP addresses. Because the match default-inspection-traffic command specifies the ports to match, any ports in the access list are ignored.

```plaintext
class-map inspection_default
    match default-inspection-traffic
policy-map type inspect dns preset_dns_map
parameters
    message-length maximum 512
policy-map global_policy
class inspection_default
    inspect dns preset_dns_map
    inspect ftp
    inspect h323 h225
    inspect h323 ras
    inspect rsh
    inspect rtsp
    inspect esmtp
    inspect sqlnet
    inspect skinny
    inspect sunrpc
    inspect xdmcp
    inspect sip
    inspect netbios
    inspect tftp
service-policy global_policy global
```
Tip

We suggest that you only inspect traffic on ports on which you expect application traffic; if you inspect all traffic, for example using `match any`, the ASASM performance can be impacted.

If you want to match non-standard ports, then create a new class map for the non-standard ports. See the “Default Settings” section on page 39-4 for the standard ports for each inspection engine. You can combine multiple class maps in the same policy if desired, so you can create one class map to match certain traffic, and another to match different traffic. However, if traffic matches a class map that contains an inspection command, and then matches another class map that also has an inspection command, only the first matching class is used. For example, SNMP matches the inspection_default class. To enable SNMP inspection, enable SNMP inspection for the default class in Step 5. Do not add another class that matches SNMP.

For example, to limit inspection to traffic from 10.1.1.0 to 192.168.1.0 using the default class map, enter the following commands:

```
hostname(config)# access-list inspect extended permit ip 10.1.1.0 255.255.255.0 192.168.1.0 255.255.255.0
hostname(config)# class-map inspection_default
hostname(config-cmap)# match access-list inspect
```

View the entire class map using the following command:

```
hostname(config-cmap)# show running-config class-map inspection_default
!
class-map inspection_default
  match default-inspection-traffic
  match access-list inspect
!
```

To inspect FTP traffic on port 21 as well as 1056 (a non-standard port), create an access list that specifies the ports, and assign it to a new class map:

```
hostname(config)# access-list ftp_inspect extended permit tcp any any eq 21
hostname(config)# access-list ftp_inspect extended permit tcp any any eq 1056
hostname(config)# class-map new_inspection
hostname(config-cmap)# match access-list ftp_inspect
```

Step 2

(Optional) Some inspection engines let you control additional parameters when you apply the inspection to the traffic. See the following sections to configure an inspection policy map for your application:

- **DCERPC**—See the “Configuring a DCERPC Inspection Policy Map for Additional Inspection Control” section on page 43-2
- **DNS**—See the “Configuring a DNS Inspection Policy Map for Additional Inspection Control” section on page 40-7
- **ESMTP**—See the “Configuring an ESMTP Inspection Policy Map for Additional Inspection Control” section on page 40-32
- **FTP**—See the “Configuring an FTP Inspection Policy Map for Additional Inspection Control” section on page 40-12.
- **GTP**—See the “Configuring a GTP Inspection Policy Map for Additional Inspection Control” section on page 43-4.
- **H323**—See the “Configuring an H.323 Inspection Policy Map for Additional Inspection Control” section on page 41-6
- **HTTP**—See the “Configuring an HTTP Inspection Policy Map for Additional Inspection Control” section on page 40-17.
• Instant Messaging—See the “Configuring an Instant Messaging Inspection Policy Map for Additional Inspection Control” section on page 40-21
• IP Options—See the “Configuring an IP Options Inspection Policy Map for Additional Inspection Control” section on page 40-25
• MGCP—See the “Configuring an MGCP Inspection Policy Map for Additional Inspection Control” section on page 41-13.
• NetBIOS—See the “Configuring a NetBIOS Inspection Policy Map for Additional Inspection Control” section on page 40-29
• RADIUS Accounting—See the “Configuring a RADIUS Inspection Policy Map for Additional Inspection Control” section on page 43-10
• RTSP—See the “Configuring an RTSP Inspection Policy Map for Additional Inspection Control” section on page 41-16
• SIP—See the “Configuring a SIP Inspection Policy Map for Additional Inspection Control” section on page 41-20
• Skinny—See the “Configuring a Skinny (SCCP) Inspection Policy Map for Additional Inspection Control” section on page 41-26
• SNMP—See the “Configuring an SNMP Inspection Policy Map for Additional Inspection Control” section on page 43-11.

Step 3
To add or edit a Layer 3/4 policy map that sets the actions to take with the class map traffic, enter the following command:

```
hostname(config)# policy-map name
hostname(config-pmap)#
```

The default policy map is called “global_policy.” This policy map includes the default inspections listed in the “Default Settings” section on page 39-4. If you want to modify the default policy (for example, to add or delete an inspection, or to identify an additional class map for your actions), then enter `global_policy` as the name.

Step 4
To identify the class map from Step 1 to which you want to assign an action, enter the following command:

```
hostname(config-pmap)# class class_map_name
hostname(config-pmap-c)#
```

If you are editing the default policy map, it includes the inspection_default class map. You can edit the actions for this class by entering `inspection_default` as the name. To add an additional class map to this policy map, identify a different name. You can combine multiple class maps in the same policy if desired, so you can create one class map to match certain traffic, and another to match different traffic. However, if traffic matches a class map that contains an inspection command, and then matches another class map that also has an inspection command, only the first matching class is used. For example, SNMP matches the inspection_default class map. To enable SNMP inspection, enable SNMP inspection for the default class in Step 5. Do not add another class that matches SNMP.

Step 5
Enable application inspection by entering the following command:

```
hostname(config-pmap-c)# inspect protocol
```

The `protocol` is one of the following values:
### Table 39-2  Protocol Keywords

<table>
<thead>
<tr>
<th>Keywords</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ctqbe</td>
<td>—</td>
</tr>
<tr>
<td>dcerpc [map_name]</td>
<td>If you added a DCERPC inspection policy map according to “Configuring a DCERPC Inspection Policy Map for Additional Inspection Control” section on page 43-2, identify the map name in this command.</td>
</tr>
<tr>
<td>dns [map_name] [dynamic-filter-snoop]</td>
<td>If you added a DNS inspection policy map according to “Configuring a DNS Inspection Policy Map for Additional Inspection Control” section on page 40-7, identify the map name in this command. The default DNS inspection policy map name is “preset_dns_map.” The default inspection policy map sets the maximum DNS packet length to 512 bytes. To enable DNS snooping for the Botnet Traffic Filter, enter the dynamic-filter-snoop keyword. See the “Enabling DNS Snooping” section on page 46-10 for more information.</td>
</tr>
<tr>
<td>esmtp [map_name]</td>
<td>If you added an ESMTP inspection policy map according to “Configuring an ESMTP Inspection Policy Map for Additional Inspection Control” section on page 40-32, identify the map name in this command.</td>
</tr>
<tr>
<td>ftp [strict [map_name]]</td>
<td>Use the strict keyword to increase the security of protected networks by preventing web browsers from sending embedded commands in FTP requests. See the “Using the strict Option” section on page 40-11 for more information. If you added an FTP inspection policy map according to “Configuring an FTP Inspection Policy Map for Additional Inspection Control” section on page 40-12, identify the map name in this command.</td>
</tr>
<tr>
<td>gtp [map_name]</td>
<td>If you added a GTP inspection policy map according to the “Configuring a GTP Inspection Policy Map for Additional Inspection Control” section on page 43-4, identify the map name in this command.</td>
</tr>
<tr>
<td>h323 h225 [map_name]</td>
<td>If you added an H323 inspection policy map according to “Configuring an H.323 Inspection Policy Map for Additional Inspection Control” section on page 41-6, identify the map name in this command.</td>
</tr>
<tr>
<td>h323 ras [map_name]</td>
<td>If you added an H323 inspection policy map according to “Configuring an H.323 Inspection Policy Map for Additional Inspection Control” section on page 41-6, identify the map name in this command.</td>
</tr>
<tr>
<td>http [map_name]</td>
<td>If you added an HTTP inspection policy map according to the “Configuring an HTTP Inspection Policy Map for Additional Inspection Control” section on page 40-17, identify the map name in this command.</td>
</tr>
<tr>
<td>icmp</td>
<td>—</td>
</tr>
</tbody>
</table>
### Table 39-2  Protocol Keywords

<table>
<thead>
<tr>
<th>Keywords</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>icmp error</td>
<td>—</td>
</tr>
<tr>
<td>ils</td>
<td>—</td>
</tr>
<tr>
<td><strong>im</strong> [map_name]</td>
<td>If you added an Instant Messaging inspection policy map according to “Configuring an Instant Messaging Inspection Policy Map for Additional Inspection Control” section on page 40-21, identify the map name in this command.</td>
</tr>
<tr>
<td>ip-options</td>
<td>[map_name]</td>
</tr>
<tr>
<td><strong>mgcp</strong> [map_name]</td>
<td>If you added an MGCP inspection policy map according to “Configuring an MGCP Inspection Policy Map for Additional Inspection Control” section on page 41-13, identify the map name in this command.</td>
</tr>
<tr>
<td>netbios</td>
<td>[map_name]</td>
</tr>
<tr>
<td><strong>pptp</strong></td>
<td>—</td>
</tr>
<tr>
<td><strong>radius-accounting</strong> [map_name]</td>
<td>The <strong>radius-accounting</strong> keyword is only available for a management class map. See the “Creating a Layer 3/4 Class Map for Management Traffic” section on page 30-14 for more information about creating a management class map. If you added a RADIUS accounting inspection policy map according to “Configuring a RADIUS Inspection Policy Map for Additional Inspection Control” section on page 43-10, identify the map name in this command.</td>
</tr>
<tr>
<td>rsh</td>
<td>—</td>
</tr>
<tr>
<td><strong>rtsp</strong> [map_name]</td>
<td>If you added a RTSP inspection policy map according to “Configuring an RTSP Inspection Policy Map for Additional Inspection Control” section on page 41-16, identify the map name in this command.</td>
</tr>
<tr>
<td>sip</td>
<td>[map_name]</td>
</tr>
<tr>
<td><strong>skinny</strong> [map_name]</td>
<td>If you added a Skinny inspection policy map according to “Configuring a Skinny (SCCP) Inspection Policy Map for Additional Inspection Control” section on page 41-26, identify the map name in this command.</td>
</tr>
<tr>
<td>snmp</td>
<td>[map_name]</td>
</tr>
<tr>
<td></td>
<td>If you added an SNMP inspection policy map according to “Configuring an SNMP Inspection Policy Map for Additional Inspection Control” section on page 43-11, identify the map name in this command.</td>
</tr>
</tbody>
</table>
Step 6

To activate the policy map on one or more interfaces, enter the following command:

```
hostname(config)# service-policy policymap_name (global | interface interface_name)
```

Where `global` applies the policy map to all interfaces, and `interface` applies the policy to one interface. By default, the default policy map, “global_policy,” is applied globally. Only one global policy is allowed. You can override the global policy on an interface by applying a service policy to that interface. You can only apply one policy map to each interface.

---

**Table 39-2  Protocol Keywords**

<table>
<thead>
<tr>
<th>Keywords</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>sqlnet</td>
<td></td>
</tr>
<tr>
<td>sunrpc</td>
<td>The default class map includes UDP port 111; if you want to enable Sun RPC inspection for TCP port 111, you need to create a new class map that matches TCP port 111, add the class to the policy, and then apply the <code>inspect sunrpc</code> command to that class.</td>
</tr>
<tr>
<td>tftp</td>
<td></td>
</tr>
<tr>
<td>waas</td>
<td></td>
</tr>
<tr>
<td>xdmcp</td>
<td></td>
</tr>
</tbody>
</table>
Configuring Inspection of Basic Internet Protocols

This chapter describes how to configure application layer protocol inspection. Inspection engines are required for services that embed IP addressing information in the user data packet or that open secondary channels on dynamically assigned ports. These protocols require the ASASM to do a deep packet inspection instead of passing the packet through the fast path. As a result, inspection engines can affect overall throughput.

Several common inspection engines are enabled on the ASASM by default, but you might need to enable others depending on your network.

This chapter includes the following sections:

- DNS Inspection, page 40-1
- FTP Inspection, page 40-11
- HTTP Inspection, page 40-16
- ICMP Inspection, page 40-20
- ICMP Error Inspection, page 40-21
- Instant Messaging Inspection, page 40-21
- IP Options Inspection, page 40-24
- IPsec Pass Through Inspection, page 40-26
- IPv6 Inspection, page 40-27
- NetBIOS Inspection, page 40-28
- PPTP Inspection, page 40-30
- SMTP and Extended SMTP Inspection, page 40-31
- TFTP Inspection, page 40-34

DNS Inspection

This section describes DNS application inspection. This section includes the following topics:

- How DNS Application Inspection Works, page 40-2
- How DNS Rewrite Works, page 40-2
- Configuring DNS Rewrite, page 40-3
How DNS Application Inspection Works

The ASASM tears down the DNS session associated with a DNS query as soon as the DNS reply is forwarded by the ASASM. The ASASM also monitors the message exchange to ensure that the ID of the DNS reply matches the ID of the DNS query.

When DNS inspection is enabled, which is the default, the ASASM performs the following additional tasks:

- Translates the DNS record based on the configuration completed using the `alias`, `static` and `nat` commands (DNS Rewrite). Translation only applies to the A-record in the DNS reply; therefore, DNS Rewrite does not affect reverse lookups, which request the PTR record.

  **Note** DNS Rewrite is not applicable for PAT because multiple PAT rules are applicable for each A-record and the PAT rule to use is ambiguous.

- Enforces the maximum DNS message length (the default is 512 bytes and the maximum length is 65535 bytes). The ASASM performs reassembly as needed to verify that the packet length is less than the maximum length configured. The ASASM drops the packet if it exceeds the maximum length.

  **Note** If you enter the `inspect dns` command without the `maximum-length` option, DNS packet size is not checked

- Enforces a domain-name length of 255 bytes and a label length of 63 bytes.
- Verifies the integrity of the domain-name referred to by the pointer if compression pointers are encountered in the DNS message.
- Checks to see if a compression pointer loop exists.

A single connection is created for multiple DNS sessions, as long as they are between the same two hosts, and the sessions have the same 5-tuple (source/destination IP address, source/destination port, and protocol). DNS identification is tracked by `app_id`, and the idle timer for each `app_id` runs independently.

Because the `app_id` expires independently, a legitimate DNS response can only pass through the ASASM within a limited period of time and there is no resource build-up. However, if you enter the `show conn` command, you will see the idle timer of a DNS connection being reset by a new DNS session. This is due to the nature of the shared DNS connection and is by design.

How DNS Rewrite Works

When DNS inspection is enabled, DNS rewrite provides full support for NAT of DNS messages originating from any interface.

If a client on an inside network requests DNS resolution of an inside address from a DNS server on an outside interface, the DNS A-record is translated correctly. If the DNS inspection engine is disabled, the A-record is not translated.
As long as DNS inspection remains enabled, you can configure DNS rewrite using the `alias`, `static`, or `nat` commands.

For details about the configuration required see the “Configuring DNS Rewrite” section on page 40-3. DNS Rewrite performs two functions:

- Translating a public address (the routable or “mapped” address) in a DNS reply to a private address (the “real” address) when the DNS client is on a private interface.
- Translating a private address to a public address when the DNS client is on the public interface.

In Figure 40-1, the DNS server resides on the external (ISP) network. The real address of the server (192.168.100.1) has been mapped using the `static` command to the ISP-assigned address (209.165.200.5). When a web client on the inside interface attempts to access the web server with the URL `http://server.example.com`, the host running the web client sends a DNS request to the DNS server to resolve the IP address of the web server. The ASASM translates the non-routable source address in the IP header and forwards the request to the ISP network on its outside interface. When the DNS reply is returned, the ASASM applies address translation not only to the destination address, but also to the embedded IP address of the web server, which is contained in the A-record in the DNS reply. As a result, the web client on the inside network gets the correct address for connecting to the web server on the inside network.

For configuration instructions for scenarios similar to this one, see the “Configuring DNS Rewrite with Two NAT Zones” section on page 40-4.

**Figure 40-1 Translating the Address in a DNS Reply (DNS Rewrite)**

DNS rewrite also works if the client making the DNS request is on a DMZ network and the DNS server is on an inside interface. For an illustration and configuration instructions for this scenario, see the “Overview of DNS Rewrite with Three NAT Zones” section on page 40-4.

### Configuring DNS Rewrite

You configure DNS rewrite using the NAT configuration.

This section includes the following topics:

- Configuring DNS Rewrite with Two NAT Zones, page 40-4
- Overview of DNS Rewrite with Three NAT Zones, page 40-4
- Configuring DNS Rewrite with Three NAT Zones, page 40-6
Configuring DNS Rewrite with Two NAT Zones

To implement a DNS Rewrite scenario similar to the one shown in Figure 40-1, perform the following steps:

**Step 1** Create a static translation for the web server using the `dns` option. See Chapter 28, “Configuring Network Object NAT.”

**Step 2** Create an access list that permits traffic to the port that the web server listens to for HTTP requests.

   hostname(config)# access-list acl-name extended permit tcp any host mapped-address eq port

   where the arguments are as follows:
   
   acl-name—The name you give the access list.
   
   mapped-address—The translated IP address of the web server.
   
   port—The TCP port that the web server listens to for HTTP requests.

**Step 3** Apply the access list created in Step 2 to the mapped interface. To do so, use the `access-group` command, as follows:

   hostname(config)# access-group acl-name in interface mapped_ifc

**Step 4** If DNS inspection is disabled or if you want to change the maximum DNS packet length, configure DNS inspection. DNS application inspection is enabled by default with a maximum DNS packet length of 512 bytes. For configuration instructions, see the “Configuring a DNS Inspection Policy Map for Additional Inspection Control” section on page 40-7.

**Step 5** On the public DNS server, add an A-record for the web server, such as:

   domain-qualified-hostname. IN A mapped-address

   where domain-qualified-hostname is the hostname with a domain suffix, as in server.example.com. The period after the hostname is important. mapped-address is the translated IP address of the web server.

The following example configures the ASASM for the scenario shown in Figure 40-1. It assumes DNS inspection is already enabled.

   hostname(config)# object network obj-192.168.100.1-01
   hostname(config-network-object)# host 192.168.100.1
   hostname(config-network-object)# nat (inside,outside) static 209.165.200.225 dns
   hostname(config)# access-list 101 permit tcp any host 209.165.200.225 eq www
   hostname(config)# access-group 101 in interface outside

This configuration requires the following A-record on the DNS server:

   server.example.com. IN A 209.165.200.225

Overview of DNS Rewrite with Three NAT Zones

Figure 40-2 provides a more complex scenario to illustrate how DNS inspection allows NAT to operate transparently with a DNS server with minimal configuration. For configuration instructions for scenarios like this one, see the “Configuring DNS Rewrite with Three NAT Zones” section on page 40-6.
In Figure 40-2, a web server, server.example.com, has the real address 192.168.100.10 on the DMZ interface of the ASASM. A web client with the IP address 10.10.10.25 is on the inside interface and a public DNS server is on the outside interface. The site NAT policies are as follows:

- The outside DNS server holds the authoritative address record for server.example.com.
- Hosts on the outside network can contact the web server with the domain name server.example.com through the outside DNS server or with the IP address 209.165.200.5.
- Clients on the inside network can access the web server with the domain name server.example.com through the outside DNS server or with the IP address 192.168.100.10.

When a host or client on any interface accesses the DMZ web server, it queries the public DNS server for the A-record of server.example.com. The DNS server returns the A-record showing that server.example.com binds to address 209.165.200.5.

When a web client on the outside network attempts to access http://server.example.com, the sequence of events is as follows:

1. The host running the web client sends the DNS server a request for the IP address of server.example.com.
2. The DNS server responds with the IP address 209.165.200.225 in the reply.
3. The web client sends its HTTP request to 209.165.200.225.
4. The packet from the outside host reaches the ASASM at the outside interface.
5. The static rule translates the address 209.165.200.225 to 192.168.100.10 and the ASASM directs the packet to the web server on the DMZ.

When a web client on the inside network attempts to access http://server.example.com, the sequence of events is as follows:

1. The host running the web client sends the DNS server a request for the IP address of server.example.com.
2. The DNS server responds with the IP address 209.165.200.225 in the reply.
3. The ASASM receives the DNS reply and submits it to the DNS application inspection engine.

4. The DNS application inspection engine does the following:
   a. Searches for any NAT rule to undo the translation of the embedded A-record address 
      "[outside]:209.165.200.5". In this example, it finds the following static configuration:
      
      ```
      object network obj-192.168.100.10-01
      host 192.168.100.10
      nat (dmz,outside) static 209.165.200.5 dns
      ```
      
   b. Uses the static rule to rewrite the A-record as follows because the **dns** option is included:
      
      ```
      [outside]:209.165.200.225 --> [dmz]:192.168.100.10
      ```
      
      **Note** If the **dns** option were not included with the **nat** command, DNS Rewrite would not be
      performed and other processing for the packet continues.

   c. Searches for any NAT to translate the web server address, [dmz]:192.168.100.10, when
      communicating with the inside web client.
      
      No NAT rule is applicable, so application inspection completes.
      
      If a NAT rule (nat or static) were applicable, the **dns** option must also be specified. If the **dns**
      option were not specified, the A-record rewrite in step b would be reverted and other processing
      for the packet continues.

5. The ASASM sends the HTTP request to server.example.com on the DMZ interface.

### Configuring DNS Rewrite with Three NAT Zones

To enable the NAT policies for the scenario in Figure 40-2, perform the following steps:

**Step 1** Create a static translation for the web server on the DMZ network using the **dns** option. See Chapter 28, “Configuring Network Object NAT.”

**Step 2** Create an access list that permits traffic to the port that the web server listens to for HTTP requests.

```
hostname(config)# access-list acl-name extended permit tcp any host mapped-address eq port
```

where the arguments are as follows:

- **acl-name**—The name you give the access list.
- **mapped-address**—The translated IP address of the web server.
- **port**—The TCP port that the web server listens to for HTTP requests.

**Step 3** Apply the access list created in **Step 2** to the outside interface. To do so, use the **access-group** command, as follows:

```
hostname(config)# access-group acl-name in interface outside
```

**Step 4** If DNS inspection is disabled or if you want to change the maximum DNS packet length, configure DNS

inspection. DNS application inspection is enabled by default with a maximum DNS packet length of 512
bytes. For configuration instructions, see the “Configuring a DNS Inspection Policy Map for Additional

Inspection Control” section on page 40-7.

**Step 5** On the public DNS server, add an A-record for the web server, such as:

```
domain-qualified-hostname. IN A mapped-address
```
where `domain-qualified-hostname` is the hostname with a domain suffix, as in `server.example.com`. The period after the hostname is important. `mapped-address` is the translated IP address of the web server.

The following example configures the ASASM for the scenario shown in Figure 40-2. It assumes DNS inspection is already enabled.

```text
hostname(config)# object network obj-192.168.100.10-01
hostname(config-network-object)# host 192.168.100.10
hostname(config-network-object)# nat (dmz,outside) static 209.165.200.225 dns
hostname(config)# access-list 101 permit tcp any host 209.165.200.225 eq www
hostname(config)# access-group 101 in interface outside
```

This configuration requires the following A-record on the DNS server:

`server.example.com. IN A 209.165.200.225`

### Configuring a DNS Inspection Policy Map for Additional Inspection Control

DNS application inspection supports DNS message controls that provide protection against DNS spoofing and cache poisoning. User configurable rules allow filtering based on DNS header, domain name, resource record type and class. Zone transfer can be restricted between servers with this function, for example.

The Recursion Desired and Recursion Available flags in the DNS header can be masked to protect a public server from attack if that server only supports a particular internal zone. In addition, DNS randomization can be enabled avoid spoofing and cache poisoning of servers that either do not support randomization, or utilize a weak pseudo random number generator. Limiting the domain names that can be queried also restricts the domain names which can be queried, which protects the public server further.

A configurable DNS mismatch alert can be used as notification if an excessive number of mismatching DNS responses are received, which could indicate a cache poisoning attack. In addition, a configurable check to enforce a Transaction Signature be attached to all DNS messages is also supported.

To specify actions when a message violates a parameter, create a DNS inspection policy map. You can then apply the inspection policy map when you enable DNS inspection.

To create a DNS inspection policy map, perform the following steps:

1. **Step 1** (Optional) Add one or more regular expressions for use in traffic matching commands according to the “Creating a Regular Expression” section on page 12-12. See the types of text you can match in the `match` commands described in Step 3.
2. **Step 2** (Optional) Create one or more regular expression class maps to group regular expressions according to the “Creating a Regular Expression Class Map” section on page 12-15.
3. **Step 3** (Optional) Create a DNS inspection class map by performing the following steps.

   A class map groups multiple traffic matches. Traffic must match *all* of the `match` commands to match the class map. You can alternatively identify `match` commands directly in the policy map. The difference between creating a class map and defining the traffic match directly in the inspection policy map is that the class map lets you create more complex match criteria, and you can reuse class maps.

   To specify traffic that should not match the class map, use the `match not` command. For example, if the `match not` command specifies the string “example.com,” then any traffic that includes “example.com” does not match the class map.
For the traffic that you identify in this class map, you can specify actions such as drop, drop-connection, reset, mask, set the rate limit, and/or log the connection in the inspection policy map.

If you want to perform different actions for each **match** command, you should identify the traffic directly in the policy map.

a. Create the class map by entering the following command:

```
hostname(config)# class-map type inspect dns [match-all | match-any] class_map_name
```

Where **class_map_name** is the name of the class map. The **match-all** keyword is the default, and specifies that traffic must match all criteria to match the class map. The **match-any** keyword specifies that the traffic matches the class map if it matches at least one of the criteria. The CLI enters class-map configuration mode, where you can enter one or more **match** commands.

b. (Optional) To add a description to the class map, enter the following command:

```
hostname(config-cmap)# description string
```

c. (Optional) To match a specific flag that is set in the DNS header, enter the following command:

```
hostname(config-cmap)# match [not] header-flag [eq] {f_well_known | f_value}
```

Where the **f_well_known** argument is the DNS flag bit. The **f_value** argument is the 16-bit value in hex. The **eq** keyword specifies an exact match.

d. (Optional) To match a DNS type, including Query type and RR type, enter the following command:

```
hostname(config-cmap)# match [not] dns-type [eq t_well_known | t_val] {range t_val1 t_val2}
```

Where the **t_well_known** argument is the DNS flag bit. The **t_val** arguments are arbitrary values in the DNS type field (0-65535). The **range** keyword specifies a range and the **eq** keyword specifies an exact match.

e. (Optional) To match a DNS class, enter the following command:

```
hostname(config-cmap)# match [not] dns-class [eq c_well_known | c_val] {range c_val1 c_val2}
```

Where the **c_well_known** argument is the DNS class. The **c_val** arguments are arbitrary values in the DNS class field. The **range** keyword specifies a range and the **eq** keyword specifies an exact match.

f. (Optional) To match a DNS question or resource record, enter the following command:

```
hostname(config-cmap)# match {question | (resource-record answer | authority | any)}
```

Where the **question** keyword specifies the question portion of a DNS message. The **resource-record** keyword specifies the resource record portion of a DNS message. The **answer** keyword specifies the Answer RR section. The **authority** keyword specifies the Authority RR section. The **additional** keyword specifies the Additional RR section.

g. (Optional) To match a DNS message domain name list, enter the following command:

```
hostname(config-cmap)# match [not] domain-name {regex regex_id | regex class class_id}
```

The **regex regex_name** argument is the regular expression you created in Step 1. The **class regex_class_name** is the regular expression class map you created in Step 2.

**Step 4** Create a DNS inspection policy map, enter the following command:

```
hostname(config)# policy-map type inspect dns policy_map_name
```

```
hostname(config-pmap)#
```
Where the policy_map_name is the name of the policy map. The CLI enters policy-map configuration mode.

**Step 5** (Optional) To add a description to the policy map, enter the following command:

```
hostname(config-pmap)# description string
```

**Step 6** To apply actions to matching traffic, perform the following steps.

a. Specify the traffic on which you want to perform actions using one of the following methods:
   - Specify the DNS class map that you created in **Step 3** by entering the following command:
     ```
     hostname(config-pmap)# class class_map_name
     hostname(config-pmap-c)#
     ```
   - Specify traffic directly in the policy map using one of the match commands described in **Step 3**.
     If you use a match not command, then any traffic that does not match the criterion in the match not command has the action applied.

b. Specify the action you want to perform on the matching traffic by entering the following command:

   ```
   hostname(config-pmap-c)# (drop [send-protocol-error] | drop-connection [send-protocol-error] | mask | reset [log] [rate-limit message_rate])
   ```

   Not all options are available for each match or class command. See the CLI help or the command reference for the exact options available.

   The drop keyword drops all packets that match.
   The send-protocol-error keyword sends a protocol error message.
   The drop-connection keyword drops the packet and closes the connection.
   The mask keyword masks out the matching portion of the packet.
   The reset keyword drops the packet, closes the connection, and sends a TCP reset to the server and/or client.
   The log keyword, which you can use alone or with one of the other keywords, sends a system log message.

   The rate-limit message_rate argument limits the rate of messages.

You can specify multiple class or match commands in the policy map. For information about the order of class and match commands, see the “Defining Actions in an Inspection Policy Map” section on page 31-2.

**Step 7** To configure parameters that affect the inspection engine, perform the following steps:

a. To enter parameters configuration mode, enter the following command:

   ```
   hostname(config-pmap)# parameters
   hostname(config-pmap-p)#
   ```

b. To randomize the DNS identifier for a DNS query, enter the following command:

   ```
   hostname(config-pmap-p)# id-randomization
   ```

c. To enable logging for excessive DNS ID mismatches, enter the following command:

   ```
   hostname(config-pmap-p)# id-mismatch [count number duration seconds] action log
   ```

   Where the count string argument specifies the maximum number of mismatch instances before a system message log is sent. The duration seconds specifies the period, in seconds, to monitor.

d. To require a TSIG resource record to be present, enter the following command:
Chapter 40  Configuring Inspection of Basic Internet Protocols

DNS Inspection

hostname(config-pmap-p)# t.sig enforced action {drop [log] | [log]

Where the count string argument specifies the maximum number of mismatch instances before a system message log is sent. The duration seconds specifies the period, in seconds, to monitor.

The following example shows a how to define a DNS inspection policy map.

hostname(config)# regex domain_example "example\,com"
hostname(config)# regex domain_foo "foo\,com"

hostname(config)# define the domain names that the server serves
hostname(config)# class-map type inspect regex match-any my_domains
hostname(config-cmap)# match regex domain_example
hostname(config-cmap)# match regex domain_foo

hostname(config)# define a DNS map for query only
hostname(config)# class-map type inspect dns match-all pub_server_map
hostname(config-cmap)# match not header-flag QR
hostname(config-cmap)# match question
hostname(config-cmap)# match not domain-name regex class my_domains

hostname(config)# policy-map type inspect dns serv_prot
hostname(config-pmap)# class pub_server_map
hostname(config-pmap-c)# drop log
hostname(config-pmap-c)# match header-flag RD
hostname(config-pmap-c)# mask log

hostname(config)# class-map dns_serv_map
hostname(config-cmap)# match default-inspection-traffic

hostname(config)# policy-map pub_policy
hostname(config-pmap)# class dns_serv_map
hostname(config-pmap-c)# inspect dns serv_prot

hostname(config)# service-policy pub_policy interface dmz

Verifying and Monitoring DNS Inspection

To view information about the current DNS connections, enter the following command:

hostname# show conn

For connections using a DNS server, the source port of the connection may be replaced by the IP address of DNS server in the show conn command output.

A single connection is created for multiple DNS sessions, as long as they are between the same two hosts, and the sessions have the same 5-tuple (source/destination IP address, source/destination port, and protocol). DNS identification is tracked by app_id, and the idle timer for each app_id runs independently.

Because the app_id expires independently, a legitimate DNS response can only pass through the security appliance within a limited period of time and there is no resource build-up. However, when you enter the show conn command, you see the idle timer of a DNS connection being reset by a new DNS session. This is due to the nature of the shared DNS connection and is by design.

To display the statistics for DNS application inspection, enter the show service-policy command. The following is sample output from the show service-policy command:

hostname# show service-policy
Interface outside:
Service-policy: sample_policy
  Class-map: dns_port
    Inspect: dns maximum-length 1500, packet 0, drop 0, reset-drop 0

## FTP Inspection

This section describes the FTP inspection engine. This section includes the following topics:

- FTP Inspection Overview, page 40-11
- Using the strict Option, page 40-11
- Configuring an FTP Inspection Policy Map for Additional Inspection Control, page 40-12
- Verifying and Monitoring FTP Inspection, page 40-16

### FTP Inspection Overview

The FTP application inspection inspects the FTP sessions and performs four tasks:

- Prepares dynamic secondary data connection
- Tracks the FTP command-response sequence
- Generates an audit trail
- Translates the embedded IP address

FTP application inspection prepares secondary channels for FTP data transfer. Ports for these channels are negotiated through PORT or PASV commands. The channels are allocated in response to a file upload, a file download, or a directory listing event.

**Note**

If you disable FTP inspection engines with the `no inspect ftp` command, outbound users can start connections only in passive mode, and all inbound FTP is disabled.

### Using the strict Option

Using the `strict` option with the `inspect ftp` command increases the security of protected networks by preventing web browsers from sending embedded commands in FTP requests.

**Note**

To specify FTP commands that are not permitted to pass through the ASASM, create an FTP map according to the “Configuring an FTP Inspection Policy Map for Additional Inspection Control” section on page 40-12.

After you enable the `strict` option on an interface, FTP inspection enforces the following behavior:

- An FTP command must be acknowledged before the ASASM allows a new command.
- The ASASM drops connections that send embedded commands.
- The 227 and PORT commands are checked to ensure they do not appear in an error string.
Using the **strict** option may cause the failure of FTP clients that are not strictly compliant with FTP RFCs.

If the **strict** option is enabled, each FTP command and response sequence is tracked for the following anomalous activity:

- **Truncated command**—Number of commas in the PORT and PASV reply command is checked to see if it is five. If it is not five, then the PORT command is assumed to be truncated and the TCP connection is closed.
- **Incorrect command**—Checks the FTP command to see if it ends with `<CR><LF>` characters, as required by the RFC. If it does not, the connection is closed.
- **Size of RETR and STOR commands**—These are checked against a fixed constant. If the size is greater, then an error message is logged and the connection is closed.
- **Command spoofing**—The PORT command should always be sent from the client. The TCP connection is denied if a PORT command is sent from the server.
- **Reply spoofing**—PASV reply command (227) should always be sent from the server. The TCP connection is denied if a PASV reply command is sent from the client. This prevents the security hole when the user executes “227 xxxx a1, a2, a3, a4, p1, p2.”
- **TCP stream editing**—The ASASM closes the connection if it detects TCP stream editing.
- **Invalid port negotiation**—The negotiated dynamic port value is checked to see if it is less than 1024. As port numbers in the range from 1 to 1024 are reserved for well-known connections, if the negotiated port falls in this range, then the TCP connection is freed.
- **Command pipelining**—The number of characters present after the port numbers in the PORT and PASV reply command is cross checked with a constant value of 8. If it is more than 8, then the TCP connection is closed.
- **The ASASM replaces the FTP server response to the SYST command with a series of Xs. to prevent the server from revealing its system type to FTP clients. To override this default behavior, use the `no mask-syst-reply` command in the FTP map.**

### Configuring an FTP Inspection Policy Map for Additional Inspection Control

FTP command filtering and security checks are provided using strict FTP inspection for improved security and control. Protocol conformance includes packet length checks, delimiters and packet format checks, command terminator checks, and command validation.

Blocking FTP based on user values is also supported so that it is possible for FTP sites to post files for download, but restrict access to certain users. You can block FTP connections based on file type, server name, and other attributes. System message logs are generated if an FTP connection is denied after inspection.

If you want FTP inspection to allow FTP servers to reveal their system type to FTP clients, and limit the allowed FTP commands, then create and configure an FTP map. You can then apply the FTP map when you enable FTP inspection.

To create an FTP map, perform the following steps:

**Step 1** (Optional) Add one or more regular expressions for use in traffic matching commands according to the “Creating a Regular Expression” section on page 12-12. See the types of text you can match in the `match` commands described in Step 3.
Step 2  (Optional) Create one or more regular expression class maps to group regular expressions according to the “Creating a Regular Expression Class Map” section on page 12-15.

Step 3  (Optional) Create an FTP inspection class map by performing the following steps.

A class map groups multiple traffic matches. Traffic must match all of the match commands to match the class map. You can alternatively identify match commands directly in the policy map. The difference between creating a class map and defining the traffic match directly in the inspection policy map is that the class map lets you create more complex match criteria, and you can reuse class maps.

To specify traffic that should not match the class map, use the match not command. For example, if the match not command specifies the string “example.com,” then any traffic that includes “example.com” does not match the class map.

For the traffic that you identify in this class map, you can specify actions such as drop, drop-connection, reset, mask, set the rate limit, and/or log the connection in the inspection policy map.

If you want to perform different actions for each match command, you should identify the traffic directly in the policy map.

a. Create the class map by entering the following command:

```
hostname(config)# class-map type inspect ftp [match-all | match-any] class_map_name
```

Where class_map_name is the name of the class map. The match-all keyword is the default, and specifies that traffic must match all criteria to match the class map. The match-any keyword specifies that the traffic matches the class map if it matches at least one of the criteria. The CLI enters class-map configuration mode, where you can enter one or more match commands.

b. (Optional) To add a description to the class map, enter the following command:

```
hostname(config-cmap)# description string
```

c. (Optional) To match a filename for FTP transfer, enter the following command:

```
hostname(config-cmap)# match [not] filename regex [regex_name | class regex_class_name]
```

Where the regex_name is the regular expression you created in Step 1. The class regex_class_name is the regular expression class map you created in Step 2.

d. (Optional) To match a file type for FTP transfer, enter the following command:

```
hostname(config-cmap)# match [not] filetype regex [regex_name | class regex_class_name]
```

Where the regex_name is the regular expression you created in Step 1. The class regex_class_name is the regular expression class map you created in Step 2.

e. (Optional) To disallow specific FTP commands, use the following command:

```
hostname(config-cmap)# match [not] request-command ftp_command [ftp_command...]
```

Where ftp_command with one or more FTP commands that you want to restrict. See Table 40-1 for a list of the FTP commands that you can restrict.
f. (Optional) To match an FTP server, enter the following command:

```
hostname(config-cmap)# match [not] server regex [regex_name | class regex_class_name]
```

Where the \textit{regex\_name} is the regular expression you created in Step 1. The \textit{class regex\_class\_name} is the regular expression class map you created in Step 2.

g. (Optional) To match an FTP username, enter the following command:

```
hostname(config-cmap)# match [not] username regex [regex_name | class regex_class_name]
```

Where the \textit{regex\_name} is the regular expression you created in Step 1. The \textit{class regex\_class\_name} is the regular expression class map you created in Step 2.

h. (Optional) To match active FTP traffic commands PORT and EPRT, enter the following command:

```
hostname(config-cmap)# match [not] active-ftp
```

i. (Optional) To match passive FTP traffic commands PASV and EPSV, enter the following command:

```
hostname(config-cmap)# match [not] passive-ftp
```

**Step 4**
Create an FTP inspection policy map, enter the following command:

```
hostname(config)# policy-map type inspect ftp policy_map_name
hostname(config-pmap)#
```

Where the \textit{policy\_map\_name} is the name of the policy map. The CLI enters policy-map configuration mode.

**Step 5**
(Optional) To add a description to the policy map, enter the following command:

```
hostname(config-pmap)# description string
```

**Step 6**
To apply actions to matching traffic, perform the following steps.

a. Specify the traffic on which you want to perform actions using one of the following methods:

<table>
<thead>
<tr>
<th>request-command deny Option</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>appe</td>
<td>Disallows the command that appends to a file.</td>
</tr>
<tr>
<td>cdup</td>
<td>Disallows the command that changes to the parent directory of the current working directory.</td>
</tr>
<tr>
<td>dele</td>
<td>Disallows the command that deletes a file on the server.</td>
</tr>
<tr>
<td>get</td>
<td>Disallows the client command for retrieving a file from the server.</td>
</tr>
<tr>
<td>help</td>
<td>Disallows the command that provides help information.</td>
</tr>
<tr>
<td>mkd</td>
<td>Disallows the command that makes a directory on the server.</td>
</tr>
<tr>
<td>put</td>
<td>Disallows the client command for sending a file to the server.</td>
</tr>
<tr>
<td>rmd</td>
<td>Disallows the command that deletes a directory on the server.</td>
</tr>
<tr>
<td>rnfr</td>
<td>Disallows the command that specifies rename-from-filename.</td>
</tr>
<tr>
<td>rnto</td>
<td>Disallows the command that specifies rename-to-filename.</td>
</tr>
<tr>
<td>site</td>
<td>Disallows the command that are specific to the server system. Usually used for remote administration.</td>
</tr>
<tr>
<td>stou</td>
<td>Disallows the command that stores a file using a unique file name.</td>
</tr>
</tbody>
</table>
- Specify the FTP class map that you created in Step 3 by entering the following command:
  hostname(config-pmap)# class class_map_name
  hostname(config-pmap-c)#

- Specify traffic directly in the policy map using one of the match commands described in Step 3. If you use a match not command, then any traffic that does not match the criterion in the match not command has the action applied.

b. Specify the action you want to perform on the matching traffic by entering the following command:
  hostname(config-pmap-c)# { [ drop [ send-protocol-error ] | drop-connection [ send-protocol-error ] | mask | reset ] [ log ] | rate-limit message_rate }

Not all options are available for each match or class command. See the CLI help or the command reference for the exact options available.

The drop keyword drops all packets that match.
The send-protocol-error keyword sends a protocol error message.
The drop-connection keyword drops the packet and closes the connection.
The mask keyword masks out the matching portion of the packet.
The reset keyword drops the packet, closes the connection, and sends a TCP reset to the server and/or client.
The log keyword, which you can use alone or with one of the other keywords, sends a system log message.
The rate-limit message_rate argument limits the rate of messages.

You can specify multiple class or match commands in the policy map. For information about the order of class and match commands, see the “Defining Actions in an Inspection Policy Map” section on page 31-2.

Step 7
To configure parameters that affect the inspection engine, perform the following steps:

a. To enter parameters configuration mode, enter the following command:
  hostname(config-pmap)# parameters
  hostname(config-pmap-p)#

b. To mask the greeting banner from the FTP server, enter the following command:
  hostname(config-pmap-p)# mask-banner

c. To mask the reply to syst command, enter the following command:
  hostname(config-pmap-p)# mask-syst-reply

Before submitting a username and password, all FTP users are presented with a greeting banner. By default, this banner includes version information useful to hackers trying to identify weaknesses in a system. The following example shows how to mask this banner:

hostname(config)# policy-map type inspect ftp mymap
hostname(config-pmap)# parameters
hostname(config-pmap-p)# mask-banner

hostname(config)# class-map match-all ftp-traffic
hostname(config-cmap)# match port tcp eq ftp

hostname(config)# policy-map ftp-policy
hostname(config-pmap)# class ftp-traffic
hostname(config-pmap-c)# inspect ftp strict mymap
hostname(config)# service-policy ftp-policy interface inside

Verifying and Monitoring FTP Inspection

FTP application inspection generates the following log messages:
- An Audit record 303002 is generated for each file that is retrieved or uploaded.
- The FTP command is checked to see if it is RETR or STOR and the retrieve and store commands are logged.
- The username is obtained by looking up a table providing the IP address.
- The username, source IP address, destination IP address, NAT address, and the file operation are logged.
- Audit record 201005 is generated if the secondary dynamic channel preparation failed due to memory shortage.

In conjunction with NAT, the FTP application inspection translates the IP address within the application payload. This is described in detail in RFC 959.

HTTP Inspection

This section describes the HTTP inspection engine. This section includes the following topics:
- HTTP Inspection Overview, page 40-16
- Configuring an HTTP Inspection Policy Map for Additional Inspection Control, page 40-17

HTTP Inspection Overview

Use the HTTP inspection engine to protect against specific attacks and other threats that are associated with HTTP traffic. HTTP inspection performs several functions:
- Enhanced HTTP inspection
- URL screening through N2H2 or Websense
  See Information About URL Filtering, page 36-6 for information.
- Java and ActiveX filtering

The latter two features are configured in conjunction with the `filter` command. For more information about filtering, see Chapter 36, “Configuring Filtering Services.”

The enhanced HTTP inspection feature, which is also known as an application firewall and is available when you configure an HTTP map (see “Configuring an HTTP Inspection Policy Map for Additional Inspection Control”), can help prevent attackers from using HTTP messages for circumventing network security policy. It verifies the following for all HTTP messages:
- Conformance to RFC 2616
- Use of RFC-defined methods only.
- Compliance with the additional criteria.
Configuring an HTTP Inspection Policy Map for Additional Inspection Control

To specify actions when a message violates a parameter, create an HTTP inspection policy map. You can then apply the inspection policy map when you enable HTTP inspection.

Note

When you enable HTTP inspection with an inspection policy map, strict HTTP inspection with the action reset and log is enabled by default. You can change the actions performed in response to inspection failure, but you cannot disable strict inspection as long as the inspection policy map remains enabled.

To create an HTTP inspection policy map, perform the following steps:

Step 1 (Optional) Add one or more regular expressions for use in traffic matching commands according to the “Creating a Regular Expression” section on page 12-12. See the types of text you can match in the match commands described in Step 3.

Step 2 (Optional) Create one or more regular expression class maps to group regular expressions according to the “Creating a Regular Expression Class Map” section on page 12-15.

Step 3 (Optional) Create an HTTP inspection class map by performing the following steps.

A class map groups multiple traffic matches. Traffic must match all of the match commands to match the class map. You can alternatively identify match commands directly in the policy map. The difference between creating a class map and defining the traffic match directly in the inspection policy map is that the class map lets you create more complex match criteria, and you can reuse class maps.

To specify traffic that should not match the class map, use the match not command. For example, if the match not command specifies the string “example.com,” then any traffic that includes “example.com” does not match the class map.

For the traffic that you identify in this class map, you can specify actions such as drop, drop-connection, reset, mask, set the rate limit, and/or log the connection in the inspection policy map.

If you want to perform different actions for each match command, you should identify the traffic directly in the policy map.

a. Create the class map by entering the following command:

```
hostname(config)# class-map type inspect http [match-all | match-any] class_map_name
hostname(config-cmap)#
```

Where class_map_name is the name of the class map. The match-all keyword is the default, and specifies that traffic must match all criteria to match the class map. The match-any keyword specifies that the traffic matches the class map if it matches at least one of the criteria. The CLI enters class-map configuration mode, where you can enter one or more match commands.

b. (Optional) To add a description to the class map, enter the following command:

```
hostname(config-cmap)# description string
```

c. (Optional) To match traffic with a content-type field in the HTTP response that does not match the accept field in the corresponding HTTP request message, enter the following command:

```
hostname(config-cmap)# match [not] req-resp content-type mismatch
```

d. (Optional) To match text found in the HTTP request message arguments, enter the following command:

```
hostname(config-cmap)# match [not] request args regex [regex_name | class regex_class_name]
```
Where the `regex_name` is the regular expression you created in Step 1. The `class regex_class_name` is the regular expression class map you created in Step 2.

e. (Optional) To match text found in the HTTP request message body or to match traffic that exceeds the maximum HTTP request message body length, enter the following command:

```
hostname(config-cmap)# match [not] request body (regex [regex_name | class regex_class_name] | length gt max_bytes)
```

Where the `regex regex_name` argument is the regular expression you created in Step 1. The `class regex_class_name` is the regular expression class map you created in Step 2. The `length gt max_bytes` is the maximum message body length in bytes.

f. (Optional) To match text found in the HTTP request message header, or to restrict the count or length of the header, enter the following command:

```
hostname(config-cmap)# match [not] request header ([field]
[regex [regex_name | class regex_class_name]] |
[length gt max_length_bytes | count gt max_count_bytes])
```

Where the `field` is the predefined message header keyword. The `regex regex_name` argument is the regular expression you created in Step 1. The `class regex_class_name` is the regular expression class map you created in Step 2. The `length gt max_bytes` is the maximum message body length in bytes. The `count gt max_count` is the maximum number of header fields.

g. (Optional) To match text found in the HTTP request message method, enter the following command:

```
hostname(config-cmap)# match [not] request method ([method] |
[regex [regex_name | class regex_class_name]])
```

Where the `method` is the predefined message method keyword. The `regex regex_name` argument is the regular expression you created in Step 1. The `class regex_class_name` is the regular expression class map you created in Step 2.

h. (Optional) To match text found in the HTTP request message URI, enter the following command:

```
hostname(config-cmap)# match [not] request uri (regex [regex_name | class regex_class_name] | length gt max_bytes)
```

Where the `regex regex_name` argument is the regular expression you created in Step 1. The `class regex_class_name` is the regular expression class map you created in Step 2. The `length gt max_bytes` is the maximum message body length in bytes.

i. (Optional) To match text found in the HTTP response message body, or to comment out Java applet and Active X object tags in order to filter them, enter the following command:

```
hostname(config-cmap)# match [not] response body ([active-x] | [java-applet] |
[regex [regex_name | class regex_class_name] | length gt max_bytes]
```

Where the `regex regex_name` argument is the regular expression you created in Step 1. The `class regex_class_name` is the regular expression class map you created in Step 2. The `length gt max_bytes` is the maximum message body length in bytes.

j. (Optional) To match text found in the HTTP response message header, or to restrict the count or length of the header, enter the following command:

```
hostname(config-cmap)# match [not] response header ([field]
[regex [regex_name | class regex_class_name]] |
[length gt max_length_bytes | count gt max_count])
```
Where the field is the predefined message header keyword. The regex regex_name argument is the regular expression you created in Step 1. The class regex_class_name is the regular expression class map you created in Step 2. The length gt max_bytes is the maximum message body length in bytes. The count gt max_count is the maximum number of header fields.

k. (Optional) To match text found in the HTTP response message status line, enter the following command:

```
hostname(config-cmap)# match [not] response status-line (regex [regex_name | class regex_class_name])
```

Where the regex regex_name argument is the regular expression you created in Step 1. The class regex_class_name is the regular expression class map you created in Step 2.

**Step 4** Create an HTTP inspection policy map, enter the following command:

```
hostname(config)# policy-map type inspect http policy_map_name
hostname(config-pmap)#
```

Where the policy_map_name is the name of the policy map. The CLI enters policy-map configuration mode.

**Step 5** (Optional) To add a description to the policy map, enter the following command:

```
hostname(config-pmap)# description string
```

**Step 6** To apply actions to matching traffic, perform the following steps.

a. Specify the traffic on which you want to perform actions using one of the following methods:

- Specify the HTTP class map that you created in Step 3 by entering the following command:

```
hostname(config-pmap)# class class_map_name
hostname(config-pmap-c)#
```

- Specify traffic directly in the policy map using one of the match commands described in Step 3. If you use a match not command, then any traffic that does not match the criterion in the match not command has the action applied.

b. Specify the action you want to perform on the matching traffic by entering the following command:

```
hostname(config-pmap-c)# { [drop {send-protocol-error} | drop-connection {send-protocol-error} | mask | reset] {log} | rate-limit message_rate}
```

Not all options are available for each match or class command. See the CLI help or the command reference for the exact options available.

The drop keyword drops all packets that match.

The send-protocol-error keyword sends a protocol error message.

The drop-connection keyword drops the packet and closes the connection.

The mask keyword masks out the matching portion of the packet.

The reset keyword drops the packet, closes the connection, and sends a TCP reset to the server and/or client.

The log keyword, which you can use alone or with one of the other keywords, sends a system log message.

The rate-limit message_rate argument limits the rate of messages.

You can specify multiple class or match commands in the policy map. For information about the order of class and match commands, see the “Defining Actions in an Inspection Policy Map” section on page 31-2.
Step 7 To configure parameters that affect the inspection engine, perform the following steps:

a. To enter parameters configuration mode, enter the following command:

```
hostname(config-pmap)# parameters
hostname(config-pmap-p)#
```

b. To check for HTTP protocol violations, enter the following command:

```
hostname(config-pmap-p)# protocol-violation [action [drop-connection / reset / log]]
```

Where the drop-connection action closes the connection. The reset action closes the connection and sends a TCP reset to the client. The log action sends a system log message when this policy map matches traffic.

c. To substitute a string for the server header field, enter the following command:

```
hostname(config-pmap-p)# spoof-server string
```

Where the string argument is the string to substitute for the server header field. Note: WebVPN streams are not subject to the spoof-server command.

The following example shows how to define an HTTP inspection policy map that will allow and log any HTTP connection that attempts to access “www.xyz.com/.*.asp” or “www.xyz[0-9][0-9].com” with methods “GET” or “PUT.” All other URL/Method combinations will be silently allowed.

```
hostname(config)# regex url1 “www.xyz.com/.*.asp”
hostname(config)# regex url2 “www.xyz[0-9][0-9].com”
hostname(config)# regex get “GET”
hostname(config)# regex put “PUT”

hostname(config)# class-map type regex match-any url_to_log
hostname(config-cmap)# match regex url1
hostname(config-cmap)# match regex url2
hostname(config-cmap)# exit

hostname(config)# class-map type regex match-any methods_to_log
hostname(config-cmap)# match regex get
hostname(config-cmap)# match regex put
hostname(config-cmap)# exit

hostname(config)# class-map type inspect http http_url_policy
hostname(config-cmap)# match request uri regex class url_to_log
hostname(config-cmap)# match request method regex class methods_to_log
hostname(config-cmap)# exit

hostname(config)# policy-map type inspect http http_policy
hostname(config-pmap)# class http_url_policy
hostname(config-pmap-c)# log
```

ICMP Inspection

The ICMP inspection engine allows ICMP traffic to have a “session” so it can be inspected like TCP and UDP traffic. Without the ICMP inspection engine, we recommend that you do not allow ICMP through the ASASM in an access list. Without stateful inspection, ICMP can be used to attack your network. The ICMP inspection engine ensures that there is only one response for each request, and that the sequence number is correct.
ICMP Error Inspection

When this feature is enabled, the ASASM creates translation sessions for intermediate hops that send ICMP error messages, based on the NAT configuration. The ASASM overwrites the packet with the translated IP addresses.

When disabled, the ASASM does not create translation sessions for intermediate nodes that generate ICMP error messages. ICMP error messages generated by the intermediate nodes between the inside host and the ASASM reach the outside host without consuming any additional NAT resource. This is undesirable when an outside host uses the traceroute command to trace the hops to the destination on the inside of the ASASM. When the ASASM does not translate the intermediate hops, all the intermediate hops appear with the mapped destination IP address.

The ICMP payload is scanned to retrieve the five-tuple from the original packet. Using the retrieved five-tuple, a lookup is performed to determine the original address of the client. The ICMP error inspection engine makes the following changes to the ICMP packet:

- In the IP Header, the mapped IP is changed to the real IP (Destination Address) and the IP checksum is modified.
- In the ICMP Header, the ICMP checksum is modified due to the changes in the ICMP packet.
- In the Payload, the following changes are made:
  - Original packet mapped IP is changed to the real IP
  - Original packet mapped port is changed to the real Port
  - Original packet IP checksum is recalculated

Instant Messaging Inspection

This section describes the IM inspection engine. This section includes the following topics:

- IM Inspection Overview, page 40-21
- Configuring an Instant Messaging Inspection Policy Map for Additional Inspection Control, page 40-21

IM Inspection Overview

The IM inspect engine lets you apply fine grained controls on the IM application to control the network usage and stop leakage of confidential data, propagation of worms, and other threats to the corporate network.

Configuring an Instant Messaging Inspection Policy Map for Additional Inspection Control

To specify actions when a message violates a parameter, create an IM inspection policy map. You can then apply the inspection policy map when you enable IM inspection.
To create an IM inspection policy map, perform the following steps:

**Step 1**  
(Optional) Add one or more regular expressions for use in traffic matching commands according to the “Creating a Regular Expression” section on page 12-12. See the types of text you can match in the match commands described in **Step 3**.

**Step 2**  
(Optional) Create one or more regular expression class maps to group regular expressions according to the “Creating a Regular Expression Class Map” section on page 12-15.

**Step 3**  
(Optional) Create an IM inspection class map by performing the following steps.

A class map groups multiple traffic matches. Traffic must match all of the match commands to match the class map. You can alternatively identify match commands directly in the policy map. The difference between creating a class map and defining the traffic match directly in the inspection policy map is that the class map lets you create more complex match criteria, and you can reuse class maps.

To specify traffic that should not match the class map, use the match not command. For example, if the match not command specifies the string “example.com,” then any traffic that includes “example.com” does not match the class map.

For the traffic that you identify in this class map, you can specify actions such as drop-connection, reset, and/or log the connection in the inspection policy map.

If you want to perform different actions for each match command, you should identify the traffic directly in the policy map.

a. Create the class map by entering the following command:

```
hostname(config)# class-map type inspect im [match-all | match-any] class_map_name
```

Where the `class_map_name` is the name of the class map. The match-all keyword is the default, and specifies that traffic must match all criteria to match the class map. The match-any keyword specifies that the traffic matches the class map if it matches at least one of the criteria. The CLI enters class-map configuration mode, where you can enter one or more match commands.

b. (Optional) To add a description to the class map, enter the following command:

```
hostname(config-cmap)# description string
```

Where the `string` is the description of the class map (up to 200 characters).

c. (Optional) To match traffic of a specific IM protocol, such as Yahoo or MSN, enter the following command:

```
hostname(config-cmap)# match [not] protocol {im-yahoo | im-msn}
```

d. (Optional) To match a specific IM service, such as chat, file-transfer, webcam, voice-chat, conference, or games, enter the following command:

```
hostname(config-cmap)# match [not] service {chat | file-transfer | webcam | voice-chat | conference | games}
```

e. (Optional) To match the source login name of the IM message, enter the following command:

```
hostname(config-cmap)# match [not] login-name regex {class class_name | regex_name}
```

Where the `regex regex_name` argument is the regular expression you created in **Step 1**. The `class regex_class_name` argument is the regular expression class map you created in **Step 2**.

f. (Optional) To match the destination login name of the IM message, enter the following command:

```
hostname(config-cmap)# match [not] peer-login-name regex {class class_name | regex_name}
```
Where the `regex regex_name` argument is the regular expression you created in **Step 1**. The `class regex_class_name` is the regular expression class map you created in **Step 2**.

g. (Optional) To match the source IP address of the IM message, enter the following command:

```
hostname(config-cmap)# match [not] ip-address ip_address ip_address_mask
```

Where the `ip_address` and the `ip_address_mask` is the IP address and netmask of the message source.

h. (Optional) To match the destination IP address of the IM message, enter the following command:

```
hostname(config-cmap)# match [not] peer-ip-address ip_address ip_address_mask
```

Where the `ip_address` and the `ip_address_mask` is the IP address and netmask of the message destination.

i. (Optional) To match the version of the IM message, enter the following command:

```
hostname(config-cmap)# match [not] version regex {class class_name | regex_name}
```

Where the `regex regex_name` argument is the regular expression you created in **Step 1**. The `class regex_class_name` is the regular expression class map you created in **Step 2**.

j. (Optional) To match the filename of the IM message, enter the following command:

```
hostname(config-cmap)# match [not] filename regex {class class_name | regex_name}
```

Where the `regex regex_name` argument is the regular expression you created in **Step 1**. The `class regex_class_name` is the regular expression class map you created in **Step 2**.

---

**Note**

Not supported using MSN IM protocol.

---

**Step 4**

Create an IM inspection policy map, enter the following command:

```
hostname(config)# policy-map type inspect im policy_map_name
hostname(config-pmap)#
```

Where the `policy_map_name` is the name of the policy map. The CLI enters policy-map configuration mode.

**Step 5**

(Optional) To add a description to the policy map, enter the following command:

```
hostname(config-pmap)# description string
```

**Step 6**

Specify the traffic on which you want to perform actions using one of the following methods:

- Specify the IM class map that you created in **Step 3** by entering the following command:

```
hostname(config-pmap)# class class_map_name
hostname(config-pmap-c)#
```

- Specify traffic directly in the policy map using one of the `match` commands described in **Step 3**. If you use a `match not` command, then any traffic that does not match the criterion in the `match not` command has the action applied.

You can specify multiple `class` or `match` commands in the policy map. For information about the order of `class` and `match` commands, see the “Defining Actions in an Inspection Policy Map” section on page 31-2.

**Step 7**

Specify the action you want to perform on the matching traffic by entering the following command:

```
hostname(config-pmap-c)# {drop-connection | reset | log}
```
Where the **drop-connection** action closes the connection. The **reset** action closes the connection and sends a TCP reset to the client. The **log** action sends a system log message when this policy map matches traffic.

The following example shows how to define an IM inspection policy map.

```
hostname(config)# regex loginname1 "ying\@yahoo.com"
hostname(config)# regex loginname2 "Kevin\@yahoo.com"
hostname(config)# regex loginname3 "rahul\@yahoo.com"
hostname(config)# regex loginname4 "darshant\@yahoo.com"
hostname(config)# regex yahoo_version_regex "1\0"
hostname(config)# regex gif_files ".*\.gif"
hostname(config)# regex exe_files ".*\.exe"

hostname(config)# class-map type regex match-any yahoo_src_login_name_regex
hostname(config-cmap)# match regex loginname1
hostname(config-cmap)# match regex loginname2

hostname(config)# class-map type regex match-any yahoo_dst_login_name_regex
hostname(config-cmap)# match regex loginname3
hostname(config-cmap)# match regex loginname4

hostname(config)# class-map type inspect im match-any yahoo_file_block_list
hostname(config-cmap)# match filename regex gif_files
hostname(config-cmap)# match filename regex exe_files

hostname(config)# class-map type inspect im match-all yahoo_im_policy
hostname(config-cmap)# match login-name regex class yahoo_src_login_name_regex
hostname(config-cmap)# match peer-login-name regex class yahoo_dst_login_name_regex

hostname(config)# class-map type inspect im match-all yahoo_im_policy2
hostname(config-cmap)# match version regex yahoo_version_regex

hostname(config)# class-map iminspect_class_map
hostname(config-cmap)# match default-inspection-traffic

hostname(config)# policy-map type inspect im im_policy_all
hostname(config-pmap)# class yahoo_file_block_list
hostname(config-pmap-c)# match service file-transfer
hostname(config-pmap)# class yahoo_im_policy
hostname(config-pmap-c)# drop-connection
hostname(config-pmap)# class yahoo_im_policy2
hostname(config-pmap-c)# reset
hostname(config)# policy-map global_policy_name
hostname(config-pmap)# class iminspect_class_map
hostname(config-pmap-c)# inspect im im_policy_all
```

### IP Options Inspection

This section describes the IP Options inspection engine. This section includes the following topics:

- **IP Options Inspection Overview**, page 40-25
- **Configuring an IP Options Inspection Policy Map for Additional Inspection Control**, page 40-25
IP Options Inspection Overview

Each IP packet contains an IP header with the Options field. The Options field, commonly referred to as IP Options, provide for control functions that are required in some situations but unnecessary for most common communications. In particular, IP Options include provisions for time stamps, security, and special routing. Use of IP Options is optional, and the field can contain zero, one, or more options.

You can configure IP Options inspection to control which IP packets with specific IP options are allowed through the ASASM. Configuring this inspection instructs the ASASM to allow a packet to pass or to clear the specified IP options and then allow the packet to pass.

IP Options inspection can check for the following three IP options in a packet:

- **End of Options List (EOOL) or IP Option 0**—This option, which contains just a single zero byte, appears at the end of all options to mark the end of a list of options. This might not coincide with the end of the header according to the header length.

- **No Operation (NOP) or IP Option 1**—The Options field in the IP header can contain zero, one, or more options, which makes the total length of the field variable. However, the IP header must be a multiple of 32 bits. If the number of bits of all options is not a multiple of 32 bits, the NOP option is used as “internal padding” to align the options on a 32-bit boundary.

- **Router Alert (RTRALT) or IP Option 20**—This option notifies transit routers to inspect the contents of the packet even when the packet is not destined for that router. This inspection is valuable when implementing RSVP and similar protocols require relatively complex processing from the routers along the packets delivery path.

**Note**

IP Options inspection is included by default in the global inspection policy. Therefore, the ASASM allows RSVP traffic that contains packets with the Router Alert option (option 20) when the ASASM is in routed mode.

Dropping RSVP packets containing the Router Alert option can cause problems in VoIP implementations.

When you configure the ASASM to clear the Router Alert option from IP headers, the IP header changes in the following ways:

- The Options field is padded so that the field ends on a 32 bit boundary.
- Internet header length (IHL) changes.
- The total length of the packet changes.
- The checksum is recomputed.

If an IP header contains additional options other than EOOL, NOP, or RTRALT, regardless of whether the ASASM is configured to allow these options, the ASASM will drop the packet.

**Configuring an IP Options Inspection Policy Map for Additional Inspection Control**

**Step 1**

To create an IP Options inspection policy map, enter the following command:

```
hostname(config)# policy-map type inspect ip-options policy_map_name
hostname(config-pmap)#
```
Where the *policy_map_name* is the name of the policy map. The CLI enters policy-map configuration mode.

**Step 2**  
(Optional) To add a description to the policy map, enter the following command:

```
hostname(config-pmap)# description string
```

**Step 3**  
To configure parameters that affect the inspection engine, perform the following steps:

a. To enter parameters configuration mode, enter the following command:

```
hostname(config-pmap)# parameters
hostname(config-pmap-p)#
```

b. To allow or clear packets with the End of Options List (EOOL) option, enter the following command:

```
hostname(config-pmap-p)# eool action {allow | clear}
```

This option, which contains just a single zero byte, appears at the end of all options to mark the end of a list of options. This might not coincide with the end of the header according to the header length.

c. To allow or clear packets with the No Operation (NOP) option, enter the following command:

```
hostname(config-pmap-p)# nop action {allow | clear}
```

The Options field in the IP header can contain zero, one, or more options, which makes the total length of the field variable. However, the IP header must be a multiple of 32 bits. If the number of bits of all options is not a multiple of 32 bits, the NOP option is used as “internal padding” to align the options on a 32-bit boundary.

d. To allow or clear packets with the Router Alert (RTRALT) option, enter the following command:

```
hostname(config-pmap-p)# router-alert action {allow | clear}
```

This option notifies transit routers to inspect the contents of the packet even when the packet is not destined for that router. This inspection is valuable when implementing RSVP and similar protocols require relatively complex processing from the routers along the packets delivery path.

---

**Note**  
Enter the *clear* command to clear the IP option from the packet before allowing the packet through the ASASM.

---

**IPsec Pass Through Inspection**

This section describes the IPsec Pass Through inspection engine. This section includes the following topics:

- IPsec Pass Through Inspection Overview, page 40-27
- “Example for Defining an IPsec Pass Through Parameter Map” section on page 40-27
IPsec Pass Through Inspection Overview

Internet Protocol Security (IPsec) is a protocol suite for securing IP communications by authenticating and encrypting each IP packet of a data stream. IPsec also includes protocols for establishing mutual authentication between agents at the beginning of the session and negotiation of cryptographic keys to be used during the session. IPsec can be used to protect data flows between a pair of hosts (for example, computer users or servers), between a pair of security gateways (such as routers or firewalls), or between a security gateway and a host.

IPsec Pass Through application inspection provides convenient traversal of ESP (IP protocol 50) and AH (IP protocol 51) traffic associated with an IKE UDP port 500 connection. It avoids lengthy access list configuration to permit ESP and AH traffic and also provides security using timeout and max connections.

Specify IPsec Pass Through inspection parameters to identify a specific map to use for defining the parameters for the inspection. Configure a policy map for Specify IPsec Pass Through inspection to access the parameters configuration, which lets you specify the restrictions for ESP or AH traffic. You can set the per client max connections and the idle timeout in parameters configuration.

NAT and non-NAT traffic is permitted. However, PAT is not supported.

Example for Defining an IPsec Pass Through Parameter Map

The following example shows how to use access lists to identify IKE traffic, define an IPsec Pass Thru parameter map, define a policy, and apply the policy to the outside interface:

```
hostname(config)# access-list ipsecpassthruacl permit udp any any eq 500
hostname(config)# class-map ipsecpassthru-traffic
hostname(config-cmap)# match access-list ipsecpassthruacl
hostname(config)# policy-map type inspect ipsec-pass-thru iptmap
hostname(config-pmap)# parameters
hostname(config-pmap-p)# esp per-client-max 10 timeout 0:11:00
hostname(config-pmap-p)# ah per-client-max 5 timeout 0:06:00
hostname(config)# policy-map inspection_policy
hostname(config-pmap)# class ipsecpassthru-traffic
hostname(config-pmap-c)# inspect ipsec-pass-thru iptmap
hostname(config)# service-policy inspection_policy interface outside
```

IPv6 Inspection

You can configure IPv6 Inspection by using MPF rules to selectively block IPv6 traffic based on the extension header. IPv6 packets are subjected to an early security check. The ASASM always passes hop-by-hop and destination option types of extension headers while blocking router header and no next header.

You can enable default IPv6 inspection or define IPv6 inspection. By defining an MPF policy map for IPv6 inspection you can configure the ASASM to selectively drop IPv6 packets based on following types of extension headers found anywhere in the IPv6 packet:

- Hop-by-Hop Options
- Routing (Type 0)
- Fragment
- Destination Options
• Authentication
• Encapsulating Security Payload

In addition, default IPv6 inspection checks conformance to RFC 2460 for type and order of extension headers in IPv6 packets:
• IPv6 header
• Hop-by-Hop Options header (0)
• Destination Options header (60)
• Routing header (43)
• Fragment header (44)
• Authentication (51)
• Encapsulating Security Payload header (50)
• Destination Options header (60)
• No Next Header (59)

When a policy map is not configured for IPv6 inspection or a configured policy map is not associated with an interface, the ASASM drops packets with any mobility type and a routing-type IPv6 extension header that arrive at the interface.

When an IPv6 inspection policy map is created, the ASASM automatically generates a configuration to drop packets that match header routing-type in the range 0-255.

### Configuring an IPv6 Inspection Policy Map

You can configure a policy map for IPv6 inspection to handle IPv6 extension headers. The IPv6 policy map is applied to each classified IPv6 packet on the specified direction. Currently, only incoming IPv6 traffic is inspected.

---

### NetBIOS Inspection

This section describes the IM inspection engine. This section includes the following topics:

• NetBIOS Inspection Overview, page 40-28
• Configuring a NetBIOS Inspection Policy Map for Additional Inspection Control, page 40-29

### NetBIOS Inspection Overview

NetBIOS inspection is enabled by default. The NetBios inspection engine translates IP addresses in the NetBios name service (NBNS) packets according to the ASASM NAT configuration.
Configuring a NetBIOS Inspection Policy Map for Additional Inspection Control

To specify actions when a message violates a parameter, create a NETBIOS inspection policy map. You can then apply the inspection policy map when you enable NETBIOS inspection.

To create a NETBIOS inspection policy map, perform the following steps:

**Step 1** (Optional) Add one or more regular expressions for use in traffic matching commands according to the “Creating a Regular Expression” section on page 12-12. See the types of text you can match in the `match` commands described in **Step 3**.

**Step 2** (Optional) Create one or more regular expression class maps to group regular expressions according to the “Creating a Regular Expression Class Map” section on page 12-15.

**Step 3** Create a NetBIOS inspection policy map, enter the following command:

```
hostname(config)# policy-map type inspect netbios policy_map_name
```

Where the `policy_map_name` is the name of the policy map. The CLI enters policy-map configuration mode.

**Step 4** (Optional) To add a description to the policy map, enter the following command:

```
hostname(config-pmap)# description string
```

**Step 5** To apply actions to matching traffic, perform the following steps.

a. Specify the traffic on which you want to perform actions using one of the following methods:

   • Specify the NetBIOS class map that you created in **Step 3** by entering the following command:

     ```
     hostname(config-pmap)# class class_map_name
     hostname(config-pmap-c)#
     ```

   • Specify traffic directly in the policy map using one of the `match` commands described in **Step 3**. If you use a `match not` command, then any traffic that does not match the criterion in the `match not` command has the action applied.

b. Specify the action you want to perform on the matching traffic by entering the following command:

   ```
   hostname(config-pmap-c)# [drop [send-protocol-error] | drop-connection [send-protocol-error] mask reset log | rate-limit message_rate]
   ```

Not all options are available for each `match` or `class` command. See the CLI help or the command reference for the exact options available.

The `drop` keyword drops all packets that match.

The `send-protocol-error` keyword sends a protocol error message.

The `drop-connection` keyword drops the packet and closes the connection.

The `mask` keyword masks out the matching portion of the packet.

The `reset` keyword drops the packet, closes the connection, and sends a TCP reset to the server and/or client.

The `log` keyword, which you can use alone or with one of the other keywords, sends a system log message.

The `rate-limit message_rate` argument limits the rate of messages.
You can specify multiple **class** or **match** commands in the policy map. For information about the order of **class** and **match** commands, see the “Defining Actions in an Inspection Policy Map” section on page 31-2.

**Step 6**

To configure parameters that affect the inspection engine, perform the following steps:

a. To enter parameters configuration mode, enter the following command:

   ```
   hostname(config-pmap)# parameters
   hostname(config-pmap-p)#
   ```

b. To check for NETBIOS protocol violations, enter the following command:

   ```
   hostname(config-pmap-p)# protocol-violation [action [drop-connection / reset / log]]
   ```

   Where the **drop-connection** action closes the connection. The **reset** action closes the connection and sends a TCP reset to the client. The **log** action sends a system log message when this policy map matches traffic.

The following example shows how to define a NETBIOS inspection policy map.

```
hostname(config)# policy-map type inspect netbios netbios_map
hostname(config-pmap)# protocol-violation drop log

hostname(config)# policy-map netbios_policy
hostname(config-pmap)# class inspection_default
hostname(config-pmap-c)# inspect netbios netbios_map
```

---

**PPTP Inspection**

PPTP is a protocol for tunneling PPP traffic. A PPTP session is composed of one TCP channel and usually two PPTP GRE tunnels. The TCP channel is the control channel used for negotiating and managing the PPTP GRE tunnels. The GRE tunnels carries PPP sessions between the two hosts.

When enabled, PPTP application inspection inspects PPTP protocol packets and dynamically creates the GRE connections and xlates necessary to permit PPTP traffic. Only Version 1, as defined in RFC 2637, is supported.

PAT is only performed for the modified version of GRE [RFC 2637] when negotiated over the PPTP TCP control channel. Port Address Translation is *not* performed for the unmodified version of GRE [RFC 1701, RFC 1702].

Specifically, the ASASM inspects the PPTP version announcements and the outgoing call request/response sequence. Only PPTP Version 1, as defined in RFC 2637, is inspected. Further inspection on the TCP control channel is disabled if the version announced by either side is not Version 1. In addition, the outgoing-call request and reply sequence are tracked. Connections and xlates are dynamic allocated as necessary to permit subsequent secondary GRE data traffic.

The PPTP inspection engine must be enabled for PPTP traffic to be translated by PAT. Additionally, PAT is only performed for a modified version of GRE (RFC2637) and only if it is negotiated over the PPTP TCP control channel. PAT is not performed for the unmodified version of GRE (RFC 1701 and RFC 1702).

As described in RFC 2637, the PPTP protocol is mainly used for the tunneling of PPP sessions initiated from a modem bank PAC (PPTP Access Concentrator) to the headend PNS (PPTP Network Server). When used this way, the PAC is the remote client and the PNS is the server.
However, when used for VPN by Windows, the interaction is inverted. The PNS is a remote single-user PC that initiates connection to the head-end PAC to gain access to a central network.

**SMTP and Extended SMTP Inspection**

This section describes the IM inspection engine. This section includes the following topics:

- SMTP and ESMTP Inspection Overview, page 40-31
- Configuring an ESMTP Inspection Policy Map for Additional Inspection Control, page 40-32

**SMTP and ESMTP Inspection Overview**

ESMTP application inspection provides improved protection against SMTP-based attacks by restricting the types of SMTP commands that can pass through the ASASM and by adding monitoring capabilities. ESMTP is an enhancement to the SMTP protocol and is similar is most respects to SMTP. For convenience, the term SMTP is used in this document to refer to both SMTP and ESMTP. The application inspection process for extended SMTP is similar to SMTP application inspection and includes support for SMTP sessions. Most commands used in an extended SMTP session are the same as those used in an SMTP session but an ESMTP session is considerably faster and offers more options related to reliability and security, such as delivery status notification.

Extended SMTP application inspection adds support for these extended SMTP commands, including AUTH, EHLO, ETRN, HELP, SAML, SEND, SOML, STARTTLS, and VRFY. Along with the support for seven RFC 821 commands (DATA, HELO, MAIL, NOOP, QUIT, RCPT, RSET), the ASASM supports a total of fifteen SMTP commands.

Other extended SMTP commands, such as ATRN, ONEX, VERB, CHUNKING, and private extensions and are not supported. Unsupported commands are translated into Xs, which are rejected by the internal server. This results in a message such as "500 Command unknown: 'XXX'." Incomplete commands are discarded.

The ESMTP inspection engine changes the characters in the server SMTP banner to asterisks except for the "2", "0", "0" characters. Carriage return (CR) and linefeed (LF) characters are ignored.

With SMTP inspection enabled, a Telnet session used for interactive SMTP may hang if the following rules are not observed: SMTP commands must be at least four characters in length; must be terminated with carriage return and line feed; and must wait for a response before issuing the next reply.

An SMTP server responds to client requests with numeric reply codes and optional human-readable strings. SMTP application inspection controls and reduces the commands that the user can use as well as the messages that the server returns. SMTP inspection performs three primary tasks:

- Restricts SMTP requests to seven basic SMTP commands and eight extended commands.
- Monitors the SMTP command-response sequence.
- Generates an audit trail—Audit record 108002 is generated when invalid character embedded in the mail address is replaced. For more information, see RFC 821.

SMTP inspection monitors the command and response sequence for the following anomalous signatures:

- Truncated commands.
- Incorrect command termination (not terminated with <CR><LR>).
• The MAIL and RCPT commands specify who are the sender and the receiver of the mail. Mail addresses are scanned for strange characters. The pipeline character (|) is deleted (changed to a blank space) and “<”, “>” are only allowed if they are used to define a mail address (“>” must be preceded by “<”).
• Unexpected transition by the SMTP server.
• For unknown commands, the ASASM changes all the characters in the packet to X. In this case, the server generates an error code to the client. Because of the change in the packed, the TCP checksum has to be recalculated or adjusted.
• TCP stream editing.
• Command pipelining.

Configuring an ESMTP Inspection Policy Map for Additional Inspection Control

ESMTP inspection detects attacks, including spam, phishing, malformed message attacks, buffer overflow/underflow attacks. It also provides support for application security and protocol conformance, which enforce the sanity of the ESMTP messages as well as detect several attacks, block senders/receivers, and block mail relay.

To specify actions when a message violates a parameter, create an ESMTP inspection policy map. You can then apply the inspection policy map when you enable ESMTP inspection.

To create an ESMTP inspection policy map, perform the following steps:

---

**Step 1** (Optional) Add one or more regular expressions for use in traffic matching commands according to the “Creating a Regular Expression” section on page 12-12. See the types of text you can match in the `match` commands described in Step 3.

**Step 2** (Optional) Create one or more regular expression class maps to group regular expressions according to the “Creating a Regular Expression Class Map” section on page 12-15.

**Step 3** Create an ESMTP inspection policy map, enter the following command:

```
hostname(config)# policy-map type inspect esmtp policy_map_name
```

Where the `policy_map_name` is the name of the policy map. The CLI enters policy-map configuration mode.

**Step 4** (Optional) To add a description to the policy map, enter the following command:

```
hostname(config-pmap)# description string
```

**Step 5** To apply actions to matching traffic, perform the following steps.

a. Specify the traffic on which you want to perform actions using one of the following methods:

   - Specify the ESMTP class map that you created in Step 3 by entering the following command:
     ```
     hostname(config-pmap)# class class_map_name
     hostname(config-pmap-c)#
     ```
   
   - Specify traffic directly in the policy map using one of the `match` commands described in Step 3. If you use a `match not` command, then any traffic that does not match the criterion in the `match not` command has the action applied.

b. Specify the action you want to perform on the matching traffic by entering the following command:
hostname(config-pmap-c)# {
}

Not all options are available for each match or class command. See the CLI help or the command reference for the exact options available.

The drop keyword drops all packets that match.

The send-protocol-error keyword sends a protocol error message.

The drop-connection keyword drops the packet and closes the connection.

The mask keyword masks out the matching portion of the packet.

The reset keyword drops the packet, closes the connection, and sends a TCP reset to the server and/or client.

The log keyword, which you can use alone or with one of the other keywords, sends a system log message.

The rate-limit message_rate argument limits the rate of messages.

You can specify multiple class or match commands in the policy map. For information about the order of class and match commands, see the “Defining Actions in an Inspection Policy Map” section on page 31-2.

**Step 6** To configure parameters that affect the inspection engine, perform the following steps:

a. To enter parameters configuration mode, enter the following command:

   hostname(config-pmap)# parameters
   hostname(config-pmap-p)#

b. To configure a local domain name, enter the following command:

   hostname(config-pmap-p)# mail-relay domain-name action [drop-connection / log]

   Where the drop-connection action closes the connection. The log action sends a system log message when this policy map matches traffic.

c. To enforce banner obfuscation, enter the following command:

   hostname(config-pmap-p)# mask-banner

The following example shows how to define an ESMTP inspection policy map.

hostname(config)# regex user1 "user1@cisco.com"
hostname(config)# regex user2 "user2@cisco.com"
hostname(config)# regex user3 "user3@cisco.com"
hostname(config)# class-map type regex senders_black_list
hostname(config-cmap)# description "Regular expressions to filter out undesired senders"
hostname(config-cmap)# match regex user1
hostname(config-cmap)# match regex user2
hostname(config-cmap)# match regex user3

hostname(config)# policy-map type inspect esmtp advanced_esmtp_map
hostname(config-pmap)# match sender-address regex class senders_black_list
hostname(config-pmap-c)# drop-connection log

hostname(config)# policy-map outside_policy
hostname(config-pmap)# class inspection_default
hostname(config-pmap-c)# inspect esmtp advanced_esmtp_map

hostname(config)# service-policy outside_policy interface outside
TFTP Inspection

TFTP inspection is enabled by default.

TFTP, described in RFC 1350, is a simple protocol to read and write files between a TFTP server and client.

The ASASM inspects TFTP traffic and dynamically creates connections and translations, if necessary, to permit file transfer between a TFTP client and server. Specifically, the inspection engine inspects TFTP read request (RRQ), write request (WRQ), and error notification (ERROR).

A dynamic secondary channel and a PAT translation, if necessary, are allocated on a reception of a valid read (RRQ) or write (WRQ) request. This secondary channel is subsequently used by TFTP for file transfer or error notification.

Only the TFTP server can initiate traffic over the secondary channel, and at most one incomplete secondary channel can exist between the TFTP client and server. An error notification from the server closes the secondary channel.

TFTP inspection must be enabled if static PAT is used to redirect TFTP traffic.
This chapter describes how to configure application layer protocol inspection. Inspection engines are required for services that embed IP addressing information in the user data packet or that open secondary channels on dynamically assigned ports. These protocols require the ASASM to do a deep packet inspection instead of passing the packet through the fast path. As a result, inspection engines can affect overall throughput.

Several common inspection engines are enabled on the ASASM by default, but you might need to enable others depending on your network.

This chapter includes the following sections:

- **CTIQBE Inspection**, page 41-1
- **H.323 Inspection**, page 41-3
- **MGCP Inspection**, page 41-11
- **RTSP Inspection**, page 41-15
- **SIP Inspection**, page 41-19
- **Skinny (SCCP) Inspection**, page 41-25

## CTIQBE Inspection

This section describes CTIQBE application inspection. This section includes the following topics:

- **CTIQBE Inspection Overview**, page 41-1
- **Limitations and Restrictions**, page 41-2
- **Verifying and Monitoring CTIQBE Inspection**, page 41-2

### CTIQBE Inspection Overview

CTIQBE protocol inspection supports NAT, PAT, and bidirectional NAT. This enables Cisco IP SoftPhone and other Cisco TAPI/JTAPI applications to work successfully with Cisco CallManager for call setup across the ASASM.

TAPI and JTAPI are used by many Cisco VoIP applications. CTIQBE is used by Cisco TSP to communicate with Cisco CallManager.
Limitations and Restrictions

The following summarizes limitations that apply when using CTIQBE application inspection:

- CTIQBE application inspection does not support configurations with the alias command.
- Stateful failover of CTIQBE calls is not supported.
- Entering the debug ctiqbe command may delay message transmission, which may have a performance impact in a real-time environment. When you enable this debugging or logging and Cisco IP SoftPhone seems unable to complete call setup through the ASASM, increase the timeout values in the Cisco TSP settings on the system running Cisco IP SoftPhone.

The following summarizes special considerations when using CTIQBE application inspection in specific scenarios:

- If two Cisco IP SoftPhones are registered with different Cisco CallManagers, which are connected to different interfaces of the ASASM, calls between these two phones fails.
- When Cisco CallManager is located on the higher security interface compared to Cisco IP SoftPhones, if NAT or outside NAT is required for the Cisco CallManager IP address, the mapping must be static as Cisco IP SoftPhone requires the Cisco CallManager IP address to be specified explicitly in its Cisco TSP configuration on the PC.
- When using PAT or Outside PAT, if the Cisco CallManager IP address is to be translated, its TCP port 2748 must be statically mapped to the same port of the PAT (interface) address for Cisco IP SoftPhone registrations to succeed. The CTIQBE listening port (TCP 2748) is fixed and is not user-configurable on Cisco CallManager, Cisco IP SoftPhone, or Cisco TSP.

Verifying and Monitoring CTIQBE Inspection

The show ctiqbe command displays information regarding the CTIQBE sessions established across the ASASM. It shows information about the media connections allocated by the CTIQBE inspection engine.

The following is sample output from the show ctiqbe command under the following conditions. There is only one active CTIQBE session setup across the ASASM. It is established between an internal CTI device (for example, a Cisco IP SoftPhone) at local address 10.0.0.99 and an external Cisco CallManager at 172.29.1.77, where TCP port 2748 is the Cisco CallManager. The heartbeat interval for the session is 120 seconds.

```
hostname# show ctiqbe
Total: 1

<table>
<thead>
<tr>
<th>LOCAL</th>
<th>FOREIGN</th>
<th>STATE</th>
<th>HEARTBEAT</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0.0.99/1117</td>
<td>172.29.1.77/2748</td>
<td>1</td>
<td>120</td>
</tr>
</tbody>
</table>

RTP/RTCP: PAT xlates: mapped to 172.29.1.99(1028 - 1029)

MEDIA: Device ID 27 Call ID 0
Foreign 172.29.1.99 (1028 - 1029)
Local 172.29.1.88 (26822 - 26823)
```

The CTI device has already registered with the CallManager. The device internal address and RTP listening port is PATed to 172.29.1.99 UDP port 1028. Its RTCP listening port is PATed to UDP 1029.
The line beginning with **RTP/RTCP: PAT xlates:** appears only if an internal CTI device has registered with an external CallManager and the CTI device address and ports are PATed to that external interface. This line does not appear if the CallManager is located on an internal interface, or if the internal CTI device address and ports are translated to the same external interface that is used by the CallManager.

The output indicates a call has been established between this CTI device and another phone at 172.29.1.88. The RTP and RTCP listening ports of the other phone are UDP 26822 and 26823. The other phone locates on the same interface as the CallManager because the ASASM does not maintain a CTIQBE session record associated with the second phone and CallManager. The active call leg on the CTI device side can be identified with Device ID 27 and Call ID 0.

The following is sample output from the **show xlate debug** command for these CTIQBE connections:

```
hostname# show xlate debug
3 in use, 3 most used
Flags:  D - DNS, d - dump, I - identity, i - inside, n - no random,
        r - portmap, s - static
TCP PAT from inside:10.0.0.99/1117 to outside:172.29.1.99/1025 flags ri idle 0:00:22
        timeout 0:00:30
UDP PAT from inside:10.0.0.99/16908 to outside:172.29.1.99/1028 flags ri idle 0:00:00
        timeout 0:04:10
UDP PAT from inside:10.0.0.99/16909 to outside:172.29.1.99/1029 flags ri idle 0:00:23
        timeout 0:04:10
```

The **show conn state ctiqbe** command displays the status of CTIQBE connections. In the output, the media connections allocated by the CTIQBE inspection engine are denoted by a ‘C’ flag. The following is sample output from the **show conn state ctiqbe** command:

```
hostname# show conn state ctiqbe
default 1 in use, 10 most used
hostname# show conn state ctiqbe detail
default 1 in use, 10 most used
Flags:  A - awaiting inside ACK to SYN, a - awaiting outside ACK to SYN,
        B - initial SYN from outside, C - CTIQBE media, D - DNS, d - dump,
        E - outside back connection, F - outside FIN, f - inside FIN,
        G - group, g - MGCP, H - H.323, h - H.225.0, I - inbound data,
        i - incomplete, J - GTP, j - GTP data, k - Skinny media,
        M - SMTP data, m - SIP media, O - outbound data, P - inside back connection,
        q - SQL*Net data, R - outside acknowledged FIN,
        r - UDP RPC, s - awaiting outside SYN,
        T - SIP, t - SIP transient, U - up
```

**H.323 Inspection**

This section describes the H.323 application inspection. This section includes the following topics:

- **H.323 Inspection Overview, page 41-4**
- **How H.323 Works, page 41-4**
- **H.239 Support in H.245 Messages, page 41-5**
- **Limitations and Restrictions, page 41-5**
- **Configuring an H.323 Inspection Policy Map for Additional Inspection Control, page 41-6**
- **Configuring H.323 and H.225 Timeout Values, page 41-9**
- **Verifying and Monitoring H.323 Inspection, page 41-9**
H.323 Inspection Overview

H.323 inspection provides support for H.323 compliant applications such as Cisco CallManager and VocalTec Gatekeeper. H.323 is a suite of protocols defined by the International Telecommunication Union for multimedia conferences over LANs. The ASASM supports H.323 through Version 6, including H.323 v3 feature Multiple Calls on One Call Signaling Channel.

With H.323 inspection enabled, the ASASM supports multiple calls on the same call signaling channel, a feature introduced with H.323 Version 3. This feature reduces call setup time and reduces the use of ports on the ASASM.

The two major functions of H.323 inspection are as follows:

- NAT the necessary embedded IPv4 addresses in the H.225 and H.245 messages. Because H.323 messages are encoded in PER encoding format, the ASASM uses an ASN.1 decoder to decode the H.323 messages.
- Dynamically allocate the negotiated H.245 and RTP/RTCP connections.

How H.323 Works

The H.323 collection of protocols collectively may use up to two TCP connection and four to eight UDP connections. FastConnect uses only one TCP connection, and RAS uses a single UDP connection for registration, admissions, and status.

An H.323 client can initially establish a TCP connection to an H.323 server using TCP port 1720 to request Q.931 call setup. As part of the call setup process, the H.323 terminal supplies a port number to the client to use for an H.245 TCP connection. In environments where H.323 gatekeeper is in use, the initial packet is transmitted using UDP.

H.323 inspection monitors the Q.931 TCP connection to determine the H.245 port number. If the H.323 terminals are not using FastConnect, the ASASM dynamically allocates the H.245 connection based on the inspection of the H.225 messages.

The H.225 connection can also be dynamically allocated when using RAS.

Within each H.245 message, the H.323 endpoints exchange port numbers that are used for subsequent UDP data streams. H.323 inspection inspects the H.245 messages to identify these ports and dynamically creates connections for the media exchange. RTP uses the negotiated port number, while RTCP uses the next higher port number.

The H.323 control channel handles H.225 and H.245 and H.323 RAS. H.323 inspection uses the following ports:

- 1718—Gate Keeper Discovery UDP port
- 1719—RAS UDP port
- 1720—TCP Control Port

You must permit traffic for the well-known H.323 port 1719 for RAS signaling. Additionally, you must permit traffic for the well-known H.323 port 1720 for the H.225 call signaling; however, the H.245 signaling ports are negotiated between the endpoints in the H.225 signaling. When an H.323 gatekeeper is used, the ASASM opens an H.225 connection based on inspection of the ACF and RCF messages.
Chapter 41      Configuring Inspection for Voice and Video Protocols

H.323 Inspection

After inspecting the H.225 messages, the ASASM opens the H.245 channel and then inspects traffic sent over the H.245 channel as well. All H.245 messages passing through the ASASM undergo H.245 application inspection, which translates embedded IP addresses and opens the media channels negotiated in H.245 messages.

The H.323 ITU standard requires that a TPKT header, defining the length of the message, precede the H.225 and H.245, before being passed on to the reliable connection. Because the TPKT header does not necessarily need to be sent in the same TCP packet as H.225 and H.245 messages, the ASASM must remember the TPKT length to process and decode the messages properly. For each connection, the ASASM keeps a record that contains the TPKT length for the next expected message.

If the ASASM needs to perform NAT on IP addresses in messages, it changes the checksum, the UUIE length, and the TPKT, if it is included in the TCP packet with the H.225 message. If the TPKT is sent in a separate TCP packet, the ASASM proxy ACKs that TPKT and appends a new TPKT to the H.245 message with the new length.

**Note**
The ASASM does not support TCP options in the Proxy ACK for the TPKT.

Each UDP connection with a packet going through H.323 inspection is marked as an H.323 connection and times out with the H.323 timeout as configured with the `timeout` command.

**Note**
You can enable call setup between H.323 endpoints when the Gatekeeper is inside the network. The ASASM includes options to open pinholes for calls based on the RegistrationRequest/RegistrationConfirm (RRQ/RCF) messages. Because these RRQ/RCF messages are sent to and from the Gatekeeper, the calling endpoint's IP address is unknown and the ASASM opens a pinhole through source IP address/port 0/0. By default, this option is disabled. To enable call setup between H.323 endpoint, enter the `ras-rcf-pinholes enable` command during parameter configuration mode while creating an H.323 Inspection policy map. See Configuring an H.323 Inspection Policy Map for Additional Inspection Control, page 41-6.

### H.239 Support in H.245 Messages

The ASASM sits between two H.323 endpoints. When the two H.323 endpoints set up a telepresentation session so that the endpoints can send and receive a data presentation, such as spreadsheet data, the ASASM ensure successful H.239 negotiation between the endpoints.

H.239 is a standard that provides the ability for H.300 series endpoints to open an additional video channel in a single call. In a call, an endpoint (such as a video phone), sends a channel for video and a channel for data presentation. The H.239 negotiation occurs on the H.245 channel.

The ASASM opens pinholes for the additional media channel and the media control channel. The endpoints use open logical channel message (OLC) to signal a new channel creation. The message extension is part of H.245 version 13.

The decoding and encoding of the telepresentation session is enabled by default. H.239 encoding and decoding is performed by ASN.1 coder.

### Limitations and Restrictions

The following are some of the known issues and limitations when using H.323 application inspection:
Static PAT may not properly translate IP addresses embedded in optional fields within H.323 messages. If you experience this kind of problem, do not use static PAT with H.323.

H.323 application inspection is not supported with NAT between same-security-level interfaces.

When a NetMeeting client registers with an H.323 gatekeeper and tries to call an H.323 gateway that is also registered with the H.323 gatekeeper, the connection is established but no voice is heard in either direction. This problem is unrelated to the ASASM.

If you configure a network static address where the network static address is the same as a third-party netmask and address, then any outbound H.323 connection fails.

**Configuring an H.323 Inspection Policy Map for Additional Inspection Control**

To specify actions when a message violates a parameter, create an H.323 inspection policy map. You can then apply the inspection policy map when you enable H.323 inspection.

To create an H.323 inspection policy map, perform the following steps:

**Step 1** (Optional) Add one or more regular expressions for use in traffic matching commands according to the “Creating a Regular Expression” section on page 12-12. See the types of text you can match in the `match` commands described in Step 3.

**Step 2** (Optional) Create one or more regular expression class maps to group regular expressions according to the “Creating a Regular Expression Class Map” section on page 12-15.

**Step 3** (Optional) Create an H.323 inspection class map by performing the following steps.

A class map groups multiple traffic matches. Traffic must match all of the `match` commands to match the class map. You can alternatively identify `match` commands directly in the policy map. The difference between creating a class map and defining the traffic match directly in the inspection policy map is that the class map lets you create more complex match criteria, and you can reuse class maps.

To specify traffic that should not match the class map, use the `match not` command. For example, if the `match not` command specifies the string “example.com,” then any traffic that includes “example.com” does not match the class map.

For the traffic that you identify in this class map, you can specify actions such as drop-connection, reset, and/or log the connection in the inspection policy map.

If you want to perform different actions for each `match` command, you should identify the traffic directly in the policy map.

a. Create the class map by entering the following command:

```
hostname(config)# class-map type inspect h323 [match-all | match-any] class_map_name
```

Where `class_map_name` is the name of the class map. The `match-all` keyword is the default, and specifies that traffic must match all criteria to match the class map. The `match-any` keyword specifies that the traffic matches the class map if it matches at least one of the criteria. The CLI enters class-map configuration mode, where you can enter one or more `match` commands.

b. (Optional) To add a description to the class map, enter the following command:

```
hostname(config-cmap)# description string
```

Where `string` is the description of the class map (up to 200 characters).

c. (Optional) To match a called party, enter the following command:

```
hostname(config-cmap)# match [not] called-party regex {class class_name | regex_name}
```
Where the `regex regex_name` argument is the regular expression you created in Step 1. The `class regex_class_name` is the regular expression class map you created in Step 2.

d. (Optional) To match a media type, enter the following command:

```
hostname(config-cmap)# match [not] media-type {audio | data | video}
```

**Step 4** Create an H.323 inspection policy map, enter the following command:

```
hostname(config)# policy-map type inspect h323 policy_map_name
hostname(config-pmap)#
```

Where the `policy_map_name` is the name of the policy map. The CLI enters policy-map configuration mode.

**Step 5** (Optional) To add a description to the policy map, enter the following command:

```
hostname(config-pmap)# description string
```

**Step 6** To apply actions to matching traffic, perform the following steps.

a. Specify the traffic on which you want to perform actions using one of the following methods:

   - Specify the H.323 class map that you created in Step 3 by entering the following command:
     
     ```
     hostname(config-pmap-c)# class class_map_name
     hostname(config-pmap-c-c)#
     ```

   - Specify traffic directly in the policy map using one of the `match` commands described in Step 3. If you use a `match not` command, then any traffic that does not match the criterion in the `match not` command has the action applied.

b. Specify the action you want to perform on the matching traffic by entering the following command:

   ```
   ```

   Not all options are available for each `match` or `class` command. See the CLI help or the command reference for the exact options available.

   The `drop` keyword drops all packets that match.

   The `send-protocol-error` keyword sends a protocol error message.

   The `drop-connection` keyword drops the packet and closes the connection.

   The `mask` keyword masks out the matching portion of the packet.

   The `reset` keyword drops the packet, closes the connection, and sends a TCP reset to the server and/or client.

   The `log` keyword, which you can use alone or with one of the other keywords, sends a system log message.

   The `rate-limit message_rate` argument limits the rate of messages.

You can specify multiple `class` or `match` commands in the policy map. For information about the order of `class` and `match` commands, see the “Defining Actions in an Inspection Policy Map” section on page 31-2.

**Step 7** To configure parameters that affect the inspection engine, perform the following steps:

a. To enter parameters configuration mode, enter the following command:

```
hostname(config-pmap)# parameters
hostname(config-pmap-p)#
```
b. To enable call setup between H.323 endpoints, enter the following command:

```bash
hostname(config)# ras-rcf-pinholes enable
```

You can enable call setup between H.323 endpoints when the Gatekeeper is inside the network. The ASASM includes options to open pinholes for calls based on the RegistrationRequest/RegistrationConfirm (RRQ/RCF) messages. Because these RRQ/RCF messages are sent to and from the Gatekeeper, the calling endpoint's IP address is unknown and the ASASM opens a pinhole through source IP address/port 0/0. By default, this option is disabled.

c. To define the H.323 call duration limit, enter the following command:

```bash
hostname(config-pmap-p)# call-duration-limit time
```

Where `time` is the call duration limit in seconds. Range is from 0:0:0 to 1163:0:0. A value of 0 means never timeout.

d. To enforce call party number used in call setup, enter the following command:

```bash
hostname(config-pmap-p)# call-party-number
```

e. To enforce H.245 tunnel blocking, enter the following command:

```bash
hostname(config-pmap-p)# h245-tunnel-block action {drop-connection | log}
```

f. To define an hsi group and enter hsi group configuration mode, enter the following command:

```bash
hostname(config-pmap-p)# hsi-group id
```

Where `id` is the hsi group ID. Range is from 0 to 2147483647.

To add an hsi to the hsi group, enter the following command in hsi group configuration mode:

```bash
hostname(config-h225-map-hsi-grp)# hsi ip_address
```

Where `ip_address` is the host to add. A maximum of five hosts per hsi group are allowed.

To add an endpoint to the hsi group, enter the following command in hsi group configuration mode:

```bash
hostname(config-h225-map-hsi-grp)# endpoint ip_address if_name
```

Where `ip_address` is the endpoint to add and `if_name` is the interface through which the endpoint is connected to the security appliance. A maximum of ten endpoints per hsi group are allowed.

g. To check RTP packets flowing on the pinholes for protocol conformance, enter the following command:

```bash
hostname(config-pmap-p)# rtp-conformance [enforce-payloadtype]
```

Where the `enforce-payloadtype` keyword enforces the payload type to be audio or video based on the signaling exchange.

h. To enable state checking validation, enter the following command:

```bash
hostname(config-pmap-p)# state-checking {h225 | ras}
```

The following example shows how to configure phone number filtering:

```bash
hostname(config)# regex caller 1 "5551234567"
hostname(config)# regex caller 2 "5552345678"
hostname(config)# regex caller 3 "5553456789"
hostname(config)# class-map type inspect h323 match-all h323_traffic
```
hostname(config-pmap-c)# match called-party regex caller1
hostname(config-pmap-c)# match calling-party regex caller2

hostname(config)# policy-map type inspect h323 h323_map
hostname(config-pmap)# parameters
hostname(config-pmap-p)# class h323_traffic
hostname(config-pmap-c)# drop

## Configuring H.323 and H.225 Timeout Values

To configure the idle time after which an H.225 signalling connection is closed, use the `timeout h225` command. The default for H.225 timeout is one hour.

To configure the idle time after which an H.323 control connection is closed, use the `timeout h323` command. The default is five minutes.

## Verifying and Monitoring H.323 Inspection

This section describes how to display information about H.323 sessions. This section includes the following topics:

- Monitoring H.225 Sessions, page 41-9
- Monitoring H.245 Sessions, page 41-10
- Monitoring H.323 RAS Sessions, page 41-10

### Monitoring H.225 Sessions

The `show h225` command displays information for H.225 sessions established across the ASASM. Along with the `debug h323 h225 event`, `debug h323 h245 event`, and `show local-host` commands, this command is used for troubleshooting H.323 inspection engine issues.

Before entering the `show h225`, `show h245`, or `show h323-ras` commands, we recommend that you configure the `pager` command. If there are a lot of session records and the `pager` command is not configured, it may take a while for the `show` command output to reach its end. If there is an abnormally large number of connections, check that the sessions are timing out based on the default timeout values or the values set by you. If they are not, then there is a problem that needs to be investigated.

The following is sample output from the `show h225` command:

```
hostname# show h225
Total H.323 Calls: 1
  1 Concurrent Call(s) for
    Local: 10.130.56.3/1040  Foreign: 172.30.254.203/1720
    1. CRV 9861
    Local: 10.130.56.3/1040  Foreign: 172.30.254.203/1720
  0 Concurrent Call(s) for
    Local: 10.130.56.4/1050  Foreign: 172.30.254.205/1720
```

This output indicates that there is currently 1 active H.323 call going through the ASASM between the local endpoint 10.130.56.3 and foreign host 172.30.254.203, and for these particular endpoints, there is 1 concurrent call between them, with a CRV for that call of 9861.
For the local endpoint 10.130.56.4 and foreign host 172.30.254.205, there are 0 concurrent calls. This means that there is no active call between the endpoints even though the H.225 session still exists. This could happen if, at the time of the `show h225` command, the call has already ended but the H.225 session has not yet been deleted. Alternately, it could mean that the two endpoints still have a TCP connection opened between them because they set “maintainConnection” to TRUE, so the session is kept open until they set it to FALSE again, or until the session times out based on the H.225 timeout value in your configuration.

### Monitoring H.245 Sessions

The `show h245` command displays information for H.245 sessions established across the ASASM by endpoints using slow start. Slow start is when the two endpoints of a call open another TCP control channel for H.245. Fast start is where the H.245 messages are exchanged as part of the H.225 messages on the H.225 control channel.) Along with the `debug h323 h245 event`, `debug h323 h225 event`, and `show local-host` commands, this command is used for troubleshooting H.323 inspection engine issues.

The following is sample output from the `show h245` command:

```
hostname# show h245
Total: 1
   LOCAL           TPKT    FOREIGN         TPKT
1  10.130.56.3/1041        0       172.30.254.203/1245    0
MEDIA: LCN 258 Foreign 172.30.254.203 RTP 49608 RTCP 49609
   Local 10.130.56.3 RTP 49608 RTCP 49609
   MEDIA: LCN 259 Foreign 172.30.254.203 RTP 49606 RTCP 49607
   Local 10.130.56.3 RTP 49606 RTCP 49607
```

There is currently one H.245 control session active across the ASASM. The local endpoint is 10.130.56.3, and we are expecting the next packet from this endpoint to have a TPKT header because the TPKT value is 0. The TKTP header is a 4-byte header preceding each H.225/H.245 message. It gives the length of the message, including the 4-byte header. The foreign host endpoint is 172.30.254.203, and we are expecting the next packet from this endpoint to have a TPKT header because the TPKT value is 0.

The media negotiated between these endpoints have an LCN of 258 with the foreign RTP IP address/port pair of 172.30.254.203/49608 and an RTCP IP address/port of 172.30.254.203/49609 with a local RTP IP address/port pair of 10.130.56.3/49608 and an RTCP port of 49609.

The second LCN of 259 has a foreign RTP IP address/port pair of 172.30.254.203/49606 and an RTCP IP address/port pair of 172.30.254.203/49607 with a local RTP IP address/port pair of 10.130.56.3/49606 and RTCP port of 49607.

### Monitoring H.323 RAS Sessions

The `show h323-ras` command displays information for H.323 RAS sessions established across the ASASM between a gatekeeper and its H.323 endpoint. Along with the `debug h323 ras event` and `show local-host` commands, this command is used for troubleshooting H.323 RAS inspection engine issues.

The `show h323-ras` command displays connection information for troubleshooting H.323 inspection engine issues. The following is sample output from the `show h323-ras` command:

```
hostname# show h323-ras
Total: 1
   GK                      Caller
1  172.30.254.214 10.130.56.14
```

This output shows that there is one active registration between the gatekeeper 172.30.254.214 and its client 10.130.56.14.
MGCP Inspection

This section describes MGCP application inspection. This section includes the following topics:

- MGCP Inspection Overview, page 41-11
- Configuring an MGCP Inspection Policy Map for Additional Inspection Control, page 41-13
- Configuring MGCP Timeout Values, page 41-14
- Verifying and Monitoring MGCP Inspection, page 41-14

MGCP Inspection Overview

MGCP is a master/slave protocol used to control media gateways from external call control elements called media gateway controllers or call agents. A media gateway is typically a network element that provides conversion between the audio signals carried on telephone circuits and data packets carried over the Internet or over other packet networks. Using NAT and PAT with MGCP lets you support a large number of devices on an internal network with a limited set of external (global) addresses. Examples of media gateways are:

- Trunking gateways, that interface between the telephone network and a Voice over IP network. Such gateways typically manage a large number of digital circuits.
- Residential gateways, that provide a traditional analog (RJ11) interface to a Voice over IP network. Examples of residential gateways include cable modem/cable set-top boxes, xDSL devices, broad-band wireless devices.
- Business gateways, that provide a traditional digital PBX interface or an integrated soft PBX interface to a Voice over IP network.

Note

To avoid policy failure when upgrading from ASA version 7.1, all layer 7 and layer 3 policies must have distinct names. For instance, a previously configured policy map with the same name as a previously configured MGCP map must be changed before the upgrade.

MGCP messages are transmitted over UDP. A response is sent back to the source address (IP address and UDP port number) of the command, but the response may not arrive from the same address as the command was sent to. This can happen when multiple call agents are being used in a failover configuration and the call agent that received the command has passed control to a backup call agent, which then sends the response. Figure 41-1 illustrates how NAT can be used with MGCP.
MGCP endpoints are physical or virtual sources and destinations for data. Media gateways contain endpoints on which the call agent can create, modify and delete connections to establish and control media sessions with other multimedia endpoints. Also, the call agent can instruct the endpoints to detect certain events and generate signals. The endpoints automatically communicate changes in service state to the call agent.

MGCP transactions are composed of a command and a mandatory response. There are eight types of commands:

- CreateConnection
- ModifyConnection
- DeleteConnection
- NotificationRequest
- Notify
- AuditEndpoint
- AuditConnection
- RestartInProgress

The first four commands are sent by the call agent to the gateway. The Notify command is sent by the gateway to the call agent. The gateway may also send a DeleteConnection. The registration of the MGCP gateway with the call agent is achieved by the RestartInProgress command. The AuditEndpoint and the AuditConnection commands are sent by the call agent to the gateway.

All commands are composed of a Command header, optionally followed by a session description. All responses are composed of a Response header, optionally followed by a session description.

- The port on which the gateway receives commands from the call agent. Gateways usually listen to UDP port 2427.
• The port on which the call agent receives commands from the gateway. Call agents usually listen to UDP port 2727.

**Note**

MGCP inspection does not support the use of different IP addresses for MGCP signaling and RTP data. A common and recommended practice is to send RTP data from a resilient IP address, such as a loopback or virtual IP address; however, the ASASM requires the RTP data to come from the same address as MGCP signalling.

### Configuring an MGCP Inspection Policy Map for Additional Inspection Control

If the network has multiple call agents and gateways for which the ASASM has to open pinholes, create an MGCP map. You can then apply the MGCP map when you enable MGCP inspection.

To create an MGCP map, perform the following steps:

**Step 1**

To create an MGCP inspection policy map, enter the following command:

```
hostname(config)# policy-map type inspect mgcp map_name
```

Where the `policy_map_name` is the name of the policy map. The CLI enters policy-map configuration mode.

**Step 2**

(Optional) To add a description to the policy map, enter the following command:

```
hostname(config-pmap)# description string
```

**Step 3**

To configure parameters that affect the inspection engine, perform the following steps:

a. To enter parameters configuration mode, enter the following command:

```
hostname(config-pmap)# parameters
```

b. To configure the call agents, enter the following command for each call agent:

```
hostname(config-pmap-p)# call-agent ip_address group_id
```

Use the `call-agent` command to specify a group of call agents that can manage one or more gateways. The call agent group information is used to open connections for the call agents in the group (other than the one a gateway sends a command to) so that any of the call agents can send the response. Call agents with the same `group_id` belong to the same group. A call agent may belong to more than one group. The `group_id` option is a number from 0 to 4294967295. The `ip_address` option specifies the IP address of the call agent.

**Note**

MGCP call agents send AUEP messages to determine if MGCP end points are present. This establishes a flow through the ASASM and allows MGCP end points to register with the call agent.

c. To configure the gateways, enter the following command for each gateway:

```
hostname(config-pmap-p)# gateway ip_address group_id
```
Use the `gateway` command to specify which group of call agents are managing a particular gateway. The IP address of the gateway is specified with the `ip_address` option. The `group_id` option is a number from 0 to 4294967295 that must correspond with the `group_id` of the call agents that are managing the gateway. A gateway may only belong to one group.

d. If you want to change the maximum number of commands allowed in the MGCP command queue, enter the following command:

```
hostname(config-pmap-p)# command-queue command_limit
```

The following example shows how to define an MGCP map:

```
hostname(config)# policy-map type inspect mgcp sample_map
hostname(config-pmap)# parameters
hostname(config-pmap-p)# call-agent 10.10.11.5 101
hostname(config-pmap-p)# call-agent 10.10.11.6 101
hostname(config-pmap-p)# call-agent 10.10.11.7 102
hostname(config-pmap-p)# call-agent 10.10.11.8 102
hostname(config-pmap-p)# gateway 10.10.10.115 101
hostname(config-pmap-p)# gateway 10.10.10.116 102
hostname(config-pmap-p)# gateway 10.10.10.117 102
hostname(config-pmap-p)# command-queue 150
```

## Configuring MGCP Timeout Values

The `timeout mgcp` command lets you set the interval for inactivity after which an MGCP media connection is closed. The default is 5 minutes.

The `timeout mgcp-pat` command lets you set the timeout for PAT xlates. Because MGCP does not have a keepalive mechanism, if you use non-Cisco MGCP gateways (call agents), the PAT xlates are torn down after the default timeout interval, which is 30 seconds.

## Verifying and Monitoring MGCP Inspection

The `show mgcp commands` command lists the number of MGCP commands in the command queue. The `show mgcp sessions` command lists the number of existing MGCP sessions. The `detail` option includes additional information about each command (or session) in the output. The following is sample output from the `show mgcp commands` command:

```
hostname# show mgcp commands
1 in use, 1 most used, 200 maximum allowed
CRCX, gateway IP: host-pc-2, transaction ID: 2052, idle: 0:00:07
```

The following is sample output from the `show mgcp detail` command.

```
hostname# show mgcp commands detail
1 in use, 1 most used, 200 maximum allowed
CRCX, idle: 0:00:10
  Gateway IP       host-pc-2
  Transaction ID   2052
  Endpoint name    aaln/1
  Call ID          9876543210abcdef
  Connection ID    8123456789abcdef
  Media IP         192.168.5.7
  Media port       6058
```
The following is sample output from the `show mgcp sessions` command.

```
hostname# show mgcp sessions
1 in use, 1 most used
Gateway IP host-pc-2, connection ID 6789af54c9, active 0:00:11
```

The following is sample output from the `show mgcp sessions detail` command.

```
hostname# show mgcp sessions detail
1 in use, 1 most used
Session active 0:00:14
  Gateway IP      host-pc-2
  Call ID         9876543210abcdef
  Connection ID   6789af54c9
  Endpoint name   aaln/1
  Media lcl port  6166
  Media rmt IP    192.168.5.7
  Media rmt port  6058
```

### RTSP Inspection

This section describes RTSP application inspection. This section includes the following topics:

- RTSP Inspection Overview, page 41-15
- Using RealPlayer, page 41-16
- Restrictions and Limitations, page 41-16
- Configuring an RTSP Inspection Policy Map for Additional Inspection Control, page 41-16

### RTSP Inspection Overview

The RTSP inspection engine lets the ASASM pass RTSP packets. RTSP is used by RealAudio, RealNetworks, Apple QuickTime 4, RealPlayer, and Cisco IP/TV connections.

**Note**

For Cisco IP/TV, use RTSP TCP port 554 and TCP 8554.

RTSP applications use the well-known port 554 with TCP (rarely UDP) as a control channel. The ASASM only supports TCP, in conformity with RFC 2326. This TCP control channel is used to negotiate the data channels that is used to transmit audio/video traffic, depending on the transport mode that is configured on the client.

The supported RDT transports are: rtp/avp, rtp/avp/udp, x-real-rdt, x-real-rdt/udp, and x-pn-tng/udp.

The ASASM parses Setup response messages with a status code of 200. If the response message is travelling inbound, the server is outside relative to the ASASM and dynamic channels need to be opened for connections coming inbound from the server. If the response message is outbound, then the ASASM does not need to open dynamic channels.

Because RFC 2326 does not require that the client and server ports must be in the SETUP response message, the ASASM keeps state and remembers the client ports in the SETUP message. QuickTime places the client ports in the SETUP message and then the server responds with only the server ports.

RTSP inspection does not support PAT or dual-NAT. Also, the ASASM cannot recognize HTTP cloaking where RTSP messages are hidden in the HTTP messages.
Using RealPlayer

When using RealPlayer, it is important to properly configure transport mode. For the ASASM, add an access-list command from the server to the client or vice versa. For RealPlayer, change transport mode by clicking Options>Preferences>Transport>RTSP Settings.

If using TCP mode on the RealPlayer, select the Use TCP to Connect to Server and Attempt to use TCP for all content check boxes. On the ASASM, there is no need to configure the inspection engine.

If using UDP mode on the RealPlayer, select the Use TCP to Connect to Server and Attempt to use UDP for static content check boxes, and for live content not available via Multicast. On the ASASM, add an inspect rtsp port command.

Restrictions and Limitations

The following restrictions apply to the RTSP inspection.

- The ASASM does not support multicast RTSP or RTSP messages over UDP.
- The ASASM does not have the ability to recognize HTTP cloaking where RTSP messages are hidden in the HTTP messages.
- The ASASM cannot perform NAT on RTSP messages because the embedded IP addresses are contained in the SDP files as part of HTTP or RTSP messages. Packets could be fragmented and ASASM cannot perform NAT on fragmented packets.
- With Cisco IP/TV, the number of translates the ASASM performs on the SDP part of the message is proportional to the number of program listings in the Content Manager (each program listing can have at least six embedded IP addresses).
- You can configure NAT for Apple QuickTime 4 or RealPlayer. Cisco IP/TV only works with NAT if the Viewer and Content Manager are on the outside network and the server is on the inside network.

Configuring an RTSP Inspection Policy Map for Additional Inspection Control

To specify actions when a message violates a parameter, create an RTSP inspection policy map. You can then apply the inspection policy map when you enable RTSP inspection.

To create an RTSP inspection policy map, perform the following steps:

Step 1 (Optional) Add one or more regular expressions for use in traffic matching commands according to the “Configuring Regular Expressions” section on page 12-12. See the types of text you can match in the match commands described in Step 3.

Step 2 (Optional) Create one or more regular expression class maps to group regular expressions according to the “Creating a Regular Expression Class Map” section on page 12-15.

Step 3 (Optional) Create an RTSP inspection class map by performing the following steps.

A class map groups multiple traffic matches. Traffic must match all of the match commands to match the class map. You can alternatively identify match commands directly in the policy map. The difference between creating a class map and defining the traffic match directly in the inspection policy map is that the class map lets you create more complex match criteria, and you can reuse class maps.
To specify traffic that should not match the class map, use the `match not` command. For example, if the `match not` command specifies the string “example.com,” then any traffic that includes “example.com” does not match the class map.

For the traffic that you identify in this class map, you can specify actions such as drop-connection and/or log the connection in the inspection policy map.

If you want to perform different actions for each `match` command, you should identify the traffic directly in the policy map.

**a.** Create the class map by entering the following command:

```bash
hostname(config)# class-map type inspect rtsp [match-all | match-any] class_map_name
hostname(config-cmap)#
```

Where `class_map_name` is the name of the class map. The `match-all` keyword is the default, and specifies that traffic must match all criteria to match the class map. The `match-any` keyword specifies that the traffic matches the class map if it matches at least one of the criteria. The CLI enters class-map configuration mode, where you can enter one or more `match` commands.

**b.** (Optional) To add a description to the class map, enter the following command:

```bash
hostname(config-cmap)# description string
```

**c.** (Optional) To match an RTSP request method, enter the following command:

```bash
hostname(config-cmap)# match [not] request-method method
```

Where `method` is the type of method to match (announce, describe, get_parameter, options, pause, play, record, redirect, setup, set_parameter, teardown).

**d.** (Optional) To match URL filtering, enter the following command:

```bash
hostname(config-cmap)# match [not] url-filter regex (class class_name | regex_name)
```

Where the `regex regex_name` argument is the regular expression you created in Step 1. The `class regex_class_name` is the regular expression class map you created in Step 2.

**Step 4** To create an RTSP inspection policy map, enter the following command:

```bash
hostname(config)# policy-map type inspect rtsp policy_map_name
hostname(config-pmap)#
```

Where the `policy_map_name` is the name of the policy map. The CLI enters policy-map configuration mode.

**Step 5** (Optional) To add a description to the policy map, enter the following command:

```bash
hostname(config-pmap)# description string
```

**Step 6** To apply actions to matching traffic, perform the following steps.

**a.** Specify the traffic on which you want to perform actions using one of the following methods:

- Specify the RTSP class map that you created in Step 3 by entering the following command:

  ```bash
  hostname(config-pmap-c)# class class_map_name
  hostname(config-pmap-c-c)#
  ```

- Specify traffic directly in the policy map using one of the `match` commands described in Step 3. If you use a `match not` command, then any traffic that does not match the criterion in the `match not` command has the action applied.

**b.** Specify the action you want to perform on the matching traffic by entering the following command:

```bash
hostname(config-pmap-c-c)# {
  [drop [send-protocol-error] | drop-connection [send-protocol-error] | mask | reset] [log] | rate-limit message_rate}
```
RTSP Inspection

Not all options are available for each match or class command. See the CLI help or the command reference for the exact options available.

The drop keyword drops all packets that match.

The send-protocol-error keyword sends a protocol error message.

The drop-connection keyword drops the packet and closes the connection.

The mask keyword masks out the matching portion of the packet.

The reset keyword drops the packet, closes the connection, and sends a TCP reset to the server and/or client.

The log keyword, which you can use alone or with one of the other keywords, sends a system log message.

The rate-limit message_rate argument limits the rate of messages.

You can specify multiple class or match commands in the policy map. For information about the order of class and match commands, see the “Defining Actions in an Inspection Policy Map” section on page 31-2.

Step 7 To configure parameters that affect the inspection engine, perform the following steps:

a. To enter parameters configuration mode, enter the following command:

   hostname(config-pmap)# parameters
   hostname(config-pmap-p)#

b. To restrict usage on reserve port for media negotiation, enter the following command:

   hostname(config-pmap-p)# reserve-port-protect

c. To set the limit on the URL length allowed in the message, enter the following command:

   hostname(config-pmap-p)# url-length-limit length

   Where the length argument specifies the URL length in bytes (0 to 6000).

The following example shows a how to define an RTSP inspection policy map.

hostname(config)# regex badurl1 www.url1.com/rtsp.avi
hostname(config)# regex badurl2 www.url2.com/rtsp.rm
hostname(config)# regex badurl3 www.url3.com/rtsp.asp

hostname(config)# class-map type regex match-any badurl-list
hostname(config-cmap)# match regex badurl1
hostname(config-cmap)# match regex badurl2
hostname(config-cmap)# match regex badurl3

hostname(config)# policy-map type inspect rtsp rtsp-filter-map
hostname(config-pmap)# match url-filter regex class badurl-list
hostname(config-pmap-p)# drop-connection

hostname(config)# class-map rtsp-traffic-class
hostname(config-cmap)# match default-inspection-traffic

hostname(config)# policy-map rtsp-traffic-policy
hostname(config-pmap)# class rtsp-traffic-class
hostname(config-pmap-c)# inspect rtsp rtsp-filter-map

hostname(config)# service-policy rtsp-traffic-policy global
SIP Inspection

This section describes SIP application inspection. This section includes the following topics:

- SIP Inspection Overview, page 41-19
- SIP Instant Messaging, page 41-19
- Configuring a SIP Inspection Policy Map for Additional Inspection Control, page 41-20
- Configuring SIP Timeout Values, page 41-24
- Verifying and Monitoring SIP Inspection, page 41-24

SIP Inspection Overview

SIP, as defined by the IETF, enables call handling sessions, particularly two-party audio conferences, or “calls.” SIP works with SDP for call signalling. SDP specifies the ports for the media stream. Using SIP, the ASASM can support any SIP VoIP gateways and VoIP proxy servers. SIP and SDP are defined in the following RFCs:

- SIP: Session Initiation Protocol, RFC 3261
- SDP: Session Description Protocol, RFC 2327

To support SIP calls through the ASASM, signaling messages for the media connection addresses, media ports, and embryonic connections for the media must be inspected, because while the signaling is sent over a well-known destination port (UDP/TCP 5060), the media streams are dynamically allocated. Also, SIP embeds IP addresses in the user-data portion of the IP packet. SIP inspection applies NAT for these embedded IP addresses.

The following limitations and restrictions apply when using PAT with SIP:

- If a remote endpoint tries to register with a SIP proxy on a network protected by the ASASM, the registration fails under very specific conditions, as follows:
  - PAT is configured for the remote endpoint.
  - The SIP registrar server is on the outside network.
  - The port is missing in the contact field in the REGISTER message sent by the endpoint to the proxy server.
  - Configuring static PAT is not supported with SIP inspection. If static PAT is configured for the Cisco Unified Communications Manager, SIP inspection cannot rewrite the SIP packet. Configure one-to-one static NAT for the Cisco Unified Communications Manager.

- If a SIP device transmits a packet in which the SDP portion has an IP address in the owner/creator field (o=) that is different than the IP address in the connection field (c=), the IP address in the o= field may not be properly translated. This is due to a limitation in the SIP protocol, which does not provide a port value in the o= field.

SIP Instant Messaging

Instant Messaging refers to the transfer of messages between users in near real-time. SIP supports the Chat feature on Windows XP using Windows Messenger RTC Client version 4.7.0105 only. The MESSAGE/INFO methods and 202 Accept response are used to support IM as defined in the following RFCs:
SESSION INITIATION PROTOCOL (SIP)-SPECIFIC EVENT NOTIFICATION, RFC 3265

MESSAGE/INFO requests can come in at any time after registration/subscription. For example, two users can be online at any time, but not chat for hours. Therefore, the SIP inspection engine opens pinholes that time out according to the configured SIP timeout value. This value must be configured at least five minutes longer than the subscription duration. The subscription duration is defined in the Contact Expires value and is typically 30 minutes.

Because MESSAGE/INFO requests are typically sent using a dynamically allocated port other than port 5060, they are required to go through the SIP inspection engine.

Note

Only the Chat feature is currently supported. Whiteboard, File Transfer, and Application Sharing are not supported. RTC Client 5.0 is not supported.

SIP inspection translates the SIP text-based messages, recalculates the content length for the SDP portion of the message, and recalculates the packet length and checksum. It dynamically opens media connections for ports specified in the SDP portion of the SIP message as address/ports on which the endpoint should listen.

SIP inspection has a database with indices CALL_ID/FROM/TO from the SIP payload. These indices identify the call, the source, and the destination. This database contains the media addresses and media ports found in the SDP media information fields and the media type. There can be multiple media addresses and ports for a session. The ASASM opens RTP/RTCP connections between the two endpoints using these media addresses/ports.

The well-known port 5060 must be used on the initial call setup (INVITE) message; however, subsequent messages may not have this port number. The SIP inspection engine opens signaling connection pinholes, and marks these connections as SIP connections. This is done for the messages to reach the SIP application and be translated.

As a call is set up, the SIP session is in the “transient” state until the media address and media port is received from the called endpoint in a Response message indicating the RTP port the called endpoint listens on. If there is a failure to receive the response messages within one minute, the signaling connection is torn down.

Once the final handshake is made, the call state is moved to active and the signaling connection remains until a BYE message is received.

If an inside endpoint initiates a call to an outside endpoint, a media hole is opened to the outside interface to allow RTP/RTCP UDP packets to flow to the inside endpoint media address and media port specified in the INVITE message from the inside endpoint. Unsolicited RTP/RTCP UDP packets to an inside interface does not traverse the ASASM, unless the ASASM configuration specifically allows it.

**Configuring a SIP Inspection Policy Map for Additional Inspection Control**

To specify actions when a message violates a parameter, create a SIP inspection policy map. You can then apply the inspection policy map when you enable SIP inspection.

To create a SIP inspection policy map, perform the following steps:

**Step 1** (Optional) Add one or more regular expressions for use in traffic matching commands according to the “Configuring Regular Expressions” section on page 12-12. See the types of text you can match in the match commands described in Step 3.
Step 2  (Optional) Create one or more regular expression class maps to group regular expressions according to the “Creating a Regular Expression Class Map” section on page 12-15.

Step 3  (Optional) Create a SIP inspection class map by performing the following steps.

A class map groups multiple traffic matches. Traffic must match all of the match commands to match the class map. You can alternatively identify match commands directly in the policy map. The difference between creating a class map and defining the traffic match directly in the inspection policy map is that the class map lets you create more complex match criteria, and you can reuse class maps.

To specify traffic that should not match the class map, use the match not command. For example, if the match not command specifies the string “example.com,” then any traffic that includes “example.com” does not match the class map.

For the traffic that you identify in this class map, you can specify actions such as drop-connection, reset, and/or log the connection in the inspection policy map.

If you want to perform different actions for each match command, you should identify the traffic directly in the policy map.

a. Create the class map by entering the following command:

```
hostname(config)# class-map type inspect sip [match-all | match-any] class_map_name
```

Where the class_map_name is the name of the class map. The match-all keyword is the default, and specifies that traffic must match all criteria to match the class map. The match-any keyword specifies that the traffic matches the class map if it matches at least one of the match commands. The CLI enters class-map configuration mode, where you can enter one or more match commands.

b. (Optional) To add a description to the class map, enter the following command:

```
hostname(config-cmap)# description string
```

Where string is the description of the class map (up to 200 characters).

c. (Optional) To match a called party, as specified in the To header, enter the following command:

```
hostname(config-cmap)# match [not] called-party regex {class class_name | regex_name}
```

Where the regex regex_name argument is the regular expression you created in Step 1. The class class_regex_name is the regular expression class map you created in Step 2.

d. (Optional) To match a calling party, as specified in the From header, enter the following command:

```
hostname(config-cmap)# match [not] calling-party regex {class class_name | regex_name}
```

Where the regex regex_name argument is the regular expression you created in Step 1. The class class_regex_name is the regular expression class map you created in Step 2.

e. (Optional) To match a content length in the SIP header, enter the following command:

```
hostname(config-cmap)# match [not] content length gt length
```

Where length is the number of bytes the content length is greater than. 0 to 65536.

f. (Optional) To match an SDP content type or regular expression, enter the following command:

```
hostname(config-cmap)# match [not] content type {sdp | regex {class class_name | regex_name}}
```

Where the regex regex_name argument is the regular expression you created in Step 1. The class class_regex_name is the regular expression class map you created in Step 2.

g. (Optional) To match a SIP IM subscriber, enter the following command:

```
hostname(config-cmap)# match [not] im-subscriber regex {class class_name | regex_name}
```
Where the `regex regex_name` argument is the regular expression you created in Step 1. The class `regex_class_name` is the regular expression class map you created in Step 2.

h. (Optional) To match a SIP via header, enter the following command:

```
hostname(config-cmap)# match [not] message-path regex (class class_name | regex_name)
```

Where the `regex regex_name` argument is the regular expression you created in Step 1. The class `regex_class_name` is the regular expression class map you created in Step 2.

i. (Optional) To match a SIP request method, enter the following command:

```
hostname(config-cmap)# match [not] request-method method
```

Where `method` is the type of method to match (ack, bye, cancel, info, invite, message, notify, options, prack, refer, register, subscribe, unknown, update).

j. (Optional) To match the requester of a third-party registration, enter the following command:

```
hostname(config-cmap)# match [not] third-party-registration regex (class class_name | regex_name)
```

Where the `regex regex_name` argument is the regular expression you created in Step 1. The class `regex_class_name` is the regular expression class map you created in Step 2.

k. (Optional) To match an URI in the SIP headers, enter the following command:

```
hostname(config-cmap)# match [not] uri (sip | tel) length gt length
```

Where `length` is the number of bytes the URI is greater than. 0 to 65536.

Step 4   Create a SIP inspection policy map, enter the following command:

```
hostname(config)# policy-map type inspect sip policy_map_name
hostname(config-pmap)#
```

Where the `policy_map_name` is the name of the policy map. The CLI enters policy-map configuration mode.

Step 5   (Optional) To add a description to the policy map, enter the following command:

```
hostname(config-pmap)# description string
```

Step 6   To apply actions to matching traffic, perform the following steps.

a. Specify the traffic on which you want to perform actions using one of the following methods:

   - Specify the SIP class map that you created in Step 3 by entering the following command:
     
     ```
     hostname(config-pmap)# class class_map_name
     hostname(config-pmap-c)#
     ```

   - Specify traffic directly in the policy map using one of the `match` commands described in Step 3. If you use a `match not` command, then any traffic that does not match the criterion in the `match not` command has the action applied.

b. Specify the action you want to perform on the matching traffic by entering the following command:

   ```
   hostname(config-pmap-c)# {
   [drop [send-protocol-error] | drop-connection [send-protocol-error] | mask | reset] [log] | rate-limit message_rate} 
   ```

   Not all options are available for each `match` or `class` command. See the CLI help or the command reference for the exact options available.

   The `drop` keyword drops all packets that match.
   The `send-protocol-error` keyword sends a protocol error message.
The **drop-connection** keyword drops the packet and closes the connection.

The **mask** keyword masks out the matching portion of the packet.

The **reset** keyword drops the packet, closes the connection, and sends a TCP reset to the server and/or client.

The **log** keyword, which you can use alone or with one of the other keywords, sends a system log message.

The **rate-limit** `message_rate` argument limits the rate of messages.

You can specify multiple **class** or **match** commands in the policy map. For information about the order of **class** and **match** commands, see the “Defining Actions in an Inspection Policy Map” section on page 31-2.

**Step 7**  
To configure parameters that affect the inspection engine, perform the following steps:

a. To enter parameters configuration mode, enter the following command:

   ```
   hostname(config-pmap)# parameters
   hostname(config-pmap-p)#
   ```

b. To enable or disable instant messaging, enter the following command:

   ```
   hostname(config-pmap-p)# im
   ```

c. To enable or disable IP address privacy, enter the following command:

   ```
   hostname(config-pmap-p)# ip-address-privacy
   ```

d. To enable check on Max-forwards header field being 0 (which cannot be 0 before reaching the destination), enter the following command:

   ```
   hostname(config-pmap-p)# max-forwards-validation action {drop | drop-connection | reset | log} [log]
   ```

e. To enable check on RTP packets flowing on the pinholes for protocol conformance, enter the following command:

   ```
   hostname(config-pmap-p)# rtp-conformance [enforce-payloadtype]
   ```

   Where the **enforce-payloadtype** keyword enforces the payload type to be audio or video based on the signaling exchange.

f. To identify the Server and User-Agent header fields, which expose the software version of either a server or an endpoint, enter the following command:

   ```
   hostname(config-pmap-p)# software-version action {mask | log} [log]
   ```

   Where the **mask** keyword masks the software version in the SIP messages.

g. To enable state checking validation, enter the following command:

   ```
   hostname(config-pmap-p)# state-checking action {drop | drop-connection | reset | log} [log]
   ```

h. To enable strict verification of the header fields in the SIP messages according to RFC 3261, enter the following command:

   ```
   hostname(config-pmap-p)# strict-header-validation action {drop | drop-connection | reset | log} [log]
   ```

i. To allow non SIP traffic using the well-known SIP signaling port, enter the following command:

   ```
   hostname(config-pmap-p)# traffic-non-sip
   ```
j. To identify the non-SIP URIs present in the Alert-Info and Call-Info header fields, enter the following command:

```
hostname(config-pmap-p)# uri-non-sip action {mask | log} [log]
```

The following example shows how to disable instant messaging over SIP:

```
hostname(config)# policy-map type inspect sip mymap
hostname(config-pmap)# parameters
hostname(config-pmap-p)# no im
hostname(config)# policy-map global_policy
hostname(config-pmap)# class inspection_default
hostname(config-pmap-c)# inspect sip mymap
hostname(config)# service-policy global_policy global
```

### Configuring SIP Timeout Values

The media connections are torn down within two minutes after the connection becomes idle. This is, however, a configurable timeout and can be set for a shorter or longer period of time. To configure the timeout for the SIP control connection, enter the following command:

```
hostname(config)# timeout sip hh:mm:ss
```

This command configures the idle timeout after which a SIP control connection is closed. To configure the timeout for the SIP media connection, enter the following command:

```
hostname(config)# timeout sip_media hh:mm:ss
```

This command configures the idle timeout after which a SIP media connection is closed.

### Verifying and Monitoring SIP Inspection

The `show sip` command assists in troubleshooting SIP inspection engine issues and is described with the `inspect protocol sip udp 5060` command. The `show timeout sip` command displays the timeout value of the designated protocol.

The `show sip` command displays information for SIP sessions established across the ASASM. Along with the `debug sip` and `show local-host` commands, this command is used for troubleshooting SIP inspection engine issues.

**Note**

We recommend that you configure the `pager` command before entering the `show sip` command. If there are a lot of SIP session records and the `pager` command is not configured, it takes a while for the `show sip` command output to reach its end.

The following is sample output from the `show sip` command:

```
hostname# show sip
Total: 2
call-id c3943000-960ca-2e43-228f010.130.56.44
  state Call init, idle 0:00:01
call-id c3943000-860ca-7e1f-11f7010.130.56.45
```
This sample shows two active SIP sessions on the ASASM (as shown in the Total field). Each call-id represents a call.

The first session, with the call-id c3943000-960ca-2e43-228f@10.130.56.44, is in the state Call Init, which means the session is still in call setup. Call setup is not complete until a final response to the call has been received. For instance, the caller has already sent the INVITE, and maybe received a 100 Response, but has not yet seen the 200 OK, so the call setup is not complete yet. Any non-1xx response message is considered a final response. This session has been idle for 1 second.

The second session is in the state Active, in which call setup is complete and the endpoints are exchanging media. This session has been idle for 6 seconds.

**Skinny (SCCP) Inspection**

This section describes SCCP application inspection. This section includes the following topics:

- SCCP Inspection Overview, page 41-25
- Supporting Cisco IP Phones, page 41-26
- Restrictions and Limitations, page 41-26
- Configuring a Skinny (SCCP) Inspection Policy Map for Additional Inspection Control, page 41-26
- Verifying and Monitoring SIP Inspection, page 41-24

**SCCP Inspection Overview**

Skinny (SCCP) is a simplified protocol used in VoIP networks. Cisco IP Phones using SCCP can coexist in an H.323 environment. When used with Cisco CallManager, the SCCP client can interoperate with H.323 compliant terminals.

The ASASM supports PAT and NAT for SCCP. PAT is necessary if you have more IP phones than global IP addresses for the IP phones to use. By supporting NAT and PAT of SCCP Signaling packets, Skinny application inspection ensures that all SCCP signalling and media packets can traverse the ASASM.

Normal traffic between Cisco CallManager and Cisco IP Phones uses SCCP and is handled by SCCP inspection without any special configuration. The ASASM also supports DHCP options 150 and 66, which it accomplishes by sending the location of a TFTP server to Cisco IP Phones and other DHCP clients. Cisco IP Phones might also include DHCP option 3 in their requests, which sets the default route. For more information, see the “Using Cisco IP Phones with a DHCP Server” section on page 10-6.

**Note**

The ASASM supports inspection of traffic from Cisco IP Phones running SCCP protocol version 19 and earlier.
Skinny (SCCP) Inspection

Chapter 41   Configuring Inspection for Voice and Video Protocols

Supporting Cisco IP Phones

In topologies where Cisco CallManager is located on the higher security interface with respect to the Cisco IP Phones, if NAT is required for the Cisco CallManager IP address, the mapping must be static as a Cisco IP Phone requires the Cisco CallManager IP address to be specified explicitly in its configuration. An static identity entry allows the Cisco CallManager on the higher security interface to accept registrations from the Cisco IP Phones.

Cisco IP Phones require access to a TFTP server to download the configuration information they need to connect to the Cisco CallManager server.

When the Cisco IP Phones are on a lower security interface compared to the TFTP server, you must use an access list to connect to the protected TFTP server on UDP port 69. While you do need a static entry for the TFTP server, this does not have to be an identity static entry. When using NAT, an identity static entry maps to the same IP address. When using PAT, it maps to the same IP address and port.

When the Cisco IP Phones are on a higher security interface compared to the TFTP server and Cisco CallManager, no access list or static entry is required to allow the Cisco IP Phones to initiate the connection.

Restrictions and Limitations

The following are limitations that apply to the current version of PAT and NAT support for SCCP:

- PAT does not work with configurations containing the alias command.
- Outside NAT or PAT is not supported.

If the address of an internal Cisco CallManager is configured for NAT or PAT to a different IP address or port, registrations for external Cisco IP Phones fail because the ASASM currently does not support NAT or PAT for the file content transferred over TFTP. Although the ASASM supports NAT of TFTP messages and opens a pinhole for the TFTP file, the ASASM cannot translate the Cisco CallManager IP address and port embedded in the Cisco IP Phone configuration files that are transferred by TFTP during phone registration.

Note: The ASASM supports stateful failover of SCCP calls except for calls that are in the middle of call setup.

Configuring a Skinny (SCCP) Inspection Policy Map for Additional Inspection Control

To specify actions when a message violates a parameter, create an SCCP inspection policy map. You can then apply the inspection policy map when you enable SCCP inspection.

To create an SCCP inspection policy map, perform the following steps:

Step 1  (Optional) Add one or more regular expressions for use in traffic matching commands according to the “Configuring Regular Expressions” section on page 12-12. See the types of text you can match in the match commands described in Step 3.

Step 2  (Optional) Create one or more regular expression class maps to group regular expressions according to the “Creating a Regular Expression Class Map” section on page 12-15.

Step 3  Create an SCCP inspection policy map, enter the following command:
Skinny (SCCP) Inspection

hostname(config)# policy-map type inspect skinny policy_map_name
hostname(config-pmap)#

Where the policy_map_name is the name of the policy map. The CLI enters policy-map configuration mode.

Step 4  (Optional) To add a description to the policy map, enter the following command:
hostname(config-pmap)# description string

Step 5  To apply actions to matching traffic, perform the following steps.

a. Specify the traffic on which you want to perform actions using one of the following methods:
   • Specify the SCCP class map that you created in Step 3 by entering the following command:
     hostname(config-pmap)# class class_map_name
     hostname(config-pmap-c)#
   • Specify traffic directly in the policy map using one of the match commands described in Step 3. If you use a match not command, then any traffic that does not match the criterion in the match not command has the action applied.

b. Specify the action you want to perform on the matching traffic by entering the following command:
   hostname(config-pmap-c)# {
   [drop | send-protocol-error] |
   drop-connection [send-protocol-error] | mask | reset] [log] | rate-limit message_rate

Not all options are available for each match or class command. See the CLI help or the command reference for the exact options available.

The drop keyword drops all packets that match.
The send-protocol-error keyword sends a protocol error message.
The drop-connection keyword drops the packet and closes the connection.
The mask keyword masks out the matching portion of the packet.
The reset keyword drops the packet, closes the connection, and sends a TCP reset to the server and/or client.
The log keyword, which you can use alone or with one of the other keywords, sends a system log message.
The rate-limit message_rate argument limits the rate of messages.

Step 6  You can specify multiple class or match commands in the policy map. For information about the order of class and match commands, see the “Defining Actions in an Inspection Policy Map” section on page 31-2 . To configure parameters that affect the inspection engine, perform the following steps:

a. To enter parameters configuration mode, enter the following command:
   hostname(config-pmap-c)# parameters
   hostname(config-pmap-p)#

b. To enforce registration before calls can be placed, enter the following command:
   hostname(config-pmap-p)# enforce-registration

c. To set the maximum SCCP station message ID allowed, enter the following command:
   hostname(config-pmap-p)# message-ID max hex_value

   Where the hex_value argument is the station message ID in hex.

d. To check RTP packets flowing on the pinholes for protocol conformance, enter the following command:
Skinny (SCCP) Inspection

hostname(config-pmap-p)# rtp-conformance [enforce-payloadtype]

Where the enforce-payloadtype keyword enforces the payload type to be audio or video based on the signaling exchange.

e. To set the maximum and minimum SCCP prefix length value allowed, enter the following command:

hostname(config-pmap-p)# sccp-prefix-len {max | min} value_length

Where the value_length argument is a maximum or minimum value.

f. To configure the timeout value for signaling and media connections, enter the following command:

hostname(config-pmap-p)# timeout

The following example shows how to define an SCCP inspection policy map.

hostname(config)# policy-map type inspect skinny skinny-map
hostname(config-pmap)# parameters
hostname(config-pmap-p)# enforce-registration
hostname(config-pmap-p)# match message-id range 200 300
hostname(config-pmap-p)# drop log
hostname(config)# class-map inspection_default
hostname(config-cmap)# match default-inspection-traffic
hostname(config)# policy-map global_policy
hostname(config-pmap)# class inspection_default
hostname(config-pmap-c)# inspect skinny skinny-map
hostname(config)# service-policy global_policy global

Verifying and Monitoring SCCP Inspection

The show skinny command assists in troubleshooting SCCP (Skinny) inspection engine issues. The following is sample output from the show skinny command under the following conditions. There are two active Skinny sessions set up across the ASASM. The first one is established between an internal Cisco IP Phone at local address 10.0.0.11 and an external Cisco CallManager at 172.18.1.33. TCP port 2000 is the CallManager. The second one is established between another internal Cisco IP Phone at local address 10.0.0.22 and the same Cisco CallManager.

hostname# show skinny
LOCAL                   FOREIGN                 STATE
---------------------------------------------------------------
1       10.0.0.11/52238         172.18.1.33/2000                1
MEDIA 10.0.0.11/22948         172.18.1.22/20798
2       10.0.0.22/52232         172.18.1.33/2000                1
MEDIA 10.0.0.22/20798         172.18.1.11/22948

The output indicates that a call has been established between two internal Cisco IP Phones. The RTP listening ports of the first and second phones are UDP 22948 and 20798 respectively.

The following is sample output from the show xlate debug command for these Skinny connections:

hostname# show xlate debug
2 in use, 2 most used
Flags:  D - DNS, d - dump, I - identity, i - inside, n - no random,
       r - portmap, s - static
NAT from inside:10.0.0.11 to outside:172.18.1.11 flags si idle 0:00:16 timeout 0:05:00
NAT from inside:10.0.0.22 to outside:172.18.1.22 flags si idle 0:00:14 timeout 0:05:00
Configuring Inspection of Database and Directory Protocols

This chapter describes how to configure application layer protocol inspection. Inspection engines are required for services that embed IP addressing information in the user data packet or that open secondary channels on dynamically assigned ports. These protocols require the ASASM to do a deep packet inspection instead of passing the packet through the fast path. As a result, inspection engines can affect overall throughput.

Several common inspection engines are enabled on the ASASM by default, but you might need to enable others depending on your network.

This chapter includes the following sections:

- ILS Inspection, page 42-1
- SQL*Net Inspection, page 42-2
- Sun RPC Inspection, page 42-3

### ILS Inspection

The ILS inspection engine provides NAT support for Microsoft NetMeeting, SiteServer, and Active Directory products that use LDAP to exchange directory information with an ILS server.

The ASASM supports NAT for ILS, which is used to register and locate endpoints in the ILS or SiteServer Directory. PAT cannot be supported because only IP addresses are stored by an LDAP database.

For search responses, when the LDAP server is located outside, NAT should be considered to allow internal peers to communicate locally while registered to external LDAP servers. For such search responses, xlates are searched first, and then DNAT entries to obtain the correct address. If both of these searches fail, then the address is not changed. For sites using NAT 0 (no NAT) and not expecting DNAT interaction, we recommend that the inspection engine be turned off to provide better performance.

Additional configuration may be necessary when the ILS server is located inside the ASASM border. This would require a hole for outside clients to access the LDAP server on the specified port, typically TCP 389.

Because ILS traffic only occurs on the secondary UDP channel, the TCP connection is disconnected after the TCP inactivity interval. By default, this interval is 60 minutes and can be adjusted using the `timeout` command.
ILS/LDAP follows a client/server model with sessions handled over a single TCP connection. Depending on the client's actions, several of these sessions may be created. During connection negotiation time, a BIND PDU is sent from the client to the server. Once a successful BIND RESPONSE from the server is received, other operational messages may be exchanged (such as ADD, DEL, SEARCH, or MODIFY) to perform operations on the ILS Directory. The ADD REQUEST and SEARCH RESPONSE PDUs may contain IP addresses of NetMeeting peers, used by H.323 (SETUP and CONNECT messages) to establish the NetMeeting sessions. Microsoft NetMeeting v2.X and v3.X provides ILS support.

The ILS inspection performs the following operations:

- Decodes the LDAP REQUEST/RESPONSE PDUs using the BER decode functions
- Parses the LDAP packet
- Extracts IP addresses
- Translates IP addresses as necessary
- Encodes the PDU with translated addresses using BER encode functions
- Copies the newly encoded PDU back to the TCP packet
- Performs incremental TCP checksum and sequence number adjustment

ILS inspection has the following limitations:

- Referral requests and responses are not supported
- Users in multiple directories are not unified
- Single users having multiple identities in multiple directories cannot be recognized by NAT

**Note**

Because H225 call signalling traffic only occurs on the secondary UDP channel, the TCP connection is disconnected after the interval specified by the TCP timeout command. By default, this interval is set at 60 minutes.

---

**SQL*Net Inspection**

SQL*Net inspection is enabled by default.

The SQL*Net protocol consists of different packet types that the ASASM handles to make the data stream appear consistent to the Oracle applications on either side of the ASASM.

The default port assignment for SQL*Net is 1521. This is the value used by Oracle for SQL*Net, but this value does not agree with IANA port assignments for Structured Query Language (SQL). Use the class-map command to apply SQL*Net inspection to a range of port numbers.

**Note**

Disable SQL*Net inspection when SQL data transfer occurs on the same port as the SQL control TCP port 1521. The security appliance acts as a proxy when SQL*Net inspection is enabled and reduces the client window size from 65000 to about 16000 causing data transfer issues.

The ASASM translates all addresses and looks in the packets for all embedded ports to open for SQL*Net Version 1.

For SQL*Net Version 2, all DATA or REDIRECT packets that immediately follow REDIRECT packets with a zero data length will be fixed up.
The packets that need fix-up contain embedded host/port addresses in the following format:

\[(ADDRESS=(PROTOCOL=tcp)(DEV=6)(HOST=a.b.c.d)(PORT=a))\]

SQL*Net Version 2 TNSFrame types (Connect, Accept, Refuse, Resend, and Marker) will not be scanned for addresses to NAT nor will inspection open dynamic connections for any embedded ports in the packet.

SQL*Net Version 2 TNSFrames, Redirect, and Data packets will be scanned for ports to open and addresses to NAT, if preceded by a REDIRECT TNSFrame type with a zero data length for the payload. When the Redirect message with data length zero passes through the ASASM, a flag will be set in the connection data structure to expect the Data or Redirect message that follows to be translated and ports to be dynamically opened. If one of the TNS frames in the preceding paragraph arrive after the Redirect message, the flag will be reset.

The SQL*Net inspection engine will recalculate the checksum, change IP, TCP lengths, and readjust Sequence Numbers and Acknowledgment Numbers using the delta of the length of the new and old message.

SQL*Net Version 1 is assumed for all other cases. TNSFrame types (Connect, Accept, Refuse, Resend, Marker, Redirect, and Data) and all packets will be scanned for ports and addresses. Addresses will be translated and port connections will be opened.

Sun RPC Inspection

This section describes Sun RPC application inspection. This section includes the following topics:

- Sun RPC Inspection Overview, page 42-3
- Managing Sun RPC Services, page 42-4
- Verifying and Monitoring Sun RPC Inspection, page 42-4

Sun RPC Inspection Overview

The Sun RPC inspection engine enables or disables application inspection for the Sun RPC protocol. Sun RPC is used by NFS and NIS. Sun RPC services can run on any port. When a client attempts to access an Sun RPC service on a server, it must learn the port that service is running on. It does this by querying the port mapper process, usually rpcbind, on the well-known port of 111.

The client sends the Sun RPC program number of the service and the port mapper process responds with the port number of the service. The client sends its Sun RPC queries to the server, specifying the port identified by the port mapper process. When the server replies, the ASASM intercepts this packet and opens both embryonic TCP and UDP connections on that port.

When you configure dynamic access lists on the ASA, they are supported on the ingress direction only and the ASA drops egress traffic destined to dynamic ports. Therefore, Sun RPC inspection implements a pinhole mechanism to support egress traffic. Sun RPC inspection uses this pinhole mechanism to support outbound dynamic access lists.

To view the dynamic access lists configured for the ASASM, use the `show asp table classify domain permit` command. For information about the `show asp table classify domain permit` command, see the CLI configuration guide.
Note

Sun RPC inspection has the limitation that NAT or PAT of Sun RPC payload information is not supported.

Managing Sun RPC Services

Use the Sun RPC services table to control Sun RPC traffic through the ASASM based on established Sun RPC sessions. To create entries in the Sun RPC services table, use the `sunrpc-server` command in global configuration mode:

```
hostname(config)# sunrpc-server interface_name ip_address mask service service_type protocol (tcp | udp) port[-port] timeout hh:mm:ss
```

You can use this command to specify the timeout after which the pinhole that was opened by Sun RPC application inspection will be closed. For example, to create a timeout of 30 minutes to the Sun RPC server with the IP address 192.168.100.2, enter the following command:

```
hostname(config)# sunrpc-server inside 192.168.100.2 255.255.255.255 service 100003 protocol tcp 111 timeout 00:30:00
```

This command specifies that the pinhole that was opened by Sun RPC application inspection will be closed after 30 minutes. In this example, the Sun RPC server is on the inside interface using TCP port 111. You can also specify UDP, a different port number, or a range of ports. To specify a range of ports, separate the starting and ending port numbers in the range with a hyphen (for example, 111-113).

The service type identifies the mapping between a specific service type and the port number used for the service. To determine the service type, which in this example is 100003, use the `sunrpcinfo` command at the UNIX or Linux command line on the Sun RPC server machine.

To clear the Sun RPC configuration, enter the following command.

```
hostname(config)# clear configure sunrpc-server
```

This removes the configuration performed using the `sunrpc-server` command. The `sunrpc-server` command allows pinholes to be created with a specified timeout.

To clear the active Sun RPC services, enter the following command:

```
hostname(config)# clear sunrpc-server active
```

This clears the pinholes that are opened by Sun RPC application inspection for specific services, such as NFS or NIS.

Verifying and Monitoring Sun RPC Inspection

The sample output in this section is for a Sun RPC server with an IP address of 192.168.100.2 on the inside interface and a Sun RPC client with an IP address of 209.168.200.5 on the outside interface.

To view information about the current Sun RPC connections, enter the `show conn` command. The following is sample output from the `show conn` command:

```
hostname# show conn
15 in use, 21 most used
UDP out 209.165.200.5:800 in 192.168.100.2:2049 idle 00:00:04 flags -
UDP out 209.165.200.5:714 in 192.168.100.2:111 idle 00:00:04 flags -
UDP out 209.165.200.5:712 in 192.168.100.2:647 idle 00:00:05 flags -
UDP out 192.168.100.2:0 in 209.165.200.5:714 idle 00:00:05 flags i
hostname(config)#
```
To display the information about the Sun RPC service table configuration, enter the `show running-config sunrpc-server` command. The following is sample output from the `show running-config sunrpc-server` command:

```
hostname(config)# show running-config sunrpc-server
sunrpc-server inside 192.168.100.2 255.255.255.255 service 100003 protocol UDP port 111
timeout 0:30:00
sunrpc-server inside 192.168.100.2 255.255.255.255 service 100005 protocol UDP port 111
timeout 0:30:00
```

This output shows that a timeout interval of 30 minutes is configured on UDP port 111 for the Sun RPC server with the IP address 192.168.100.2 on the inside interface.

To display the pinholes open for Sun RPC services, enter the `show sunrpc-server active` command. The following is sample output from `show sunrpc-server active` command:

```
hostname# show sunrpc-server active
LOCAL FOREIGN SERVICE TIMEOUT
-----------------------------------------------
1 209.165.200.5/0 192.168.100.2/2049 100003 0:30:00
2 209.165.200.5/0 192.168.100.2/2049 100003 0:30:00
3 209.165.200.5/0 192.168.100.2/647 100005 0:30:00
4 209.165.200.5/0 192.168.100.2/650 100005 0:30:00
```

The entry in the LOCAL column shows the IP address of the client or server on the inside interface, while the value in the FOREIGN column shows the IP address of the client or server on the outside interface.

To view information about the Sun RPC services running on a Sun RPC server, enter the `rpcinfo -p` command from the Linux or UNIX server command line. The following is sample output from the `rpcinfo -p` command:

```
sunrpcserver:~ # rpcinfo -p
program vers proto port
100000 2 tcp 111 portmapper
100000 2 udp 111 portmapper
100024 1 udp 632 status
100024 1 tcp 632 status
100003 2 udp 2049 nfs
100003 3 udp 2049 nfs
100003 2 tcp 2049 nfs
100003 3 tcp 2049 nfs
1000021 1 udp 32771 nlockmgr
1000021 3 udp 32771 nlockmgr
1000021 4 udp 32771 nlockmgr
1000021 1 tcp 32852 nlockmgr
1000021 3 tcp 32852 nlockmgr
1000021 4 tcp 32852 nlockmgr
100005 1 udp 647 mountd
100005 1 tcp 650 mountd
100005 2 udp 647 mountd
100005 2 tcp 650 mountd
100005 3 udp 647 mountd
100005 3 tcp 650 mountd
```

In this output, port 647 corresponds to the mountd daemon running over UDP. The mountd process would more commonly be using port 32780. The mountd process running over TCP uses port 650 in this example.
Configuring Inspection for Management Application Protocols

This chapter describes how to configure application layer protocol inspection. Inspection engines are required for services that embed IP addressing information in the user data packet or that open secondary channels on dynamically assigned ports. These protocols require the ASASM to do a deep packet inspection instead of passing the packet through the fast path. As a result, inspection engines can affect overall throughput.

Several common inspection engines are enabled on the ASASM by default, but you might need to enable others depending on your network.

This chapter includes the following sections:

- DCERPC Inspection, page 43-1
- GTP Inspection, page 43-3
- RADIUS Accounting Inspection, page 43-9
- RSH Inspection, page 43-11
- SNMP Inspection, page 43-11
- XDMCP Inspection, page 43-12

DCERPC Inspection

This section describes the DCERPC inspection engine. This section includes the following topics:

- DCERPC Overview, page 43-1
- Configuring a DCERPC Inspection Policy Map for Additional Inspection Control, page 43-2

DCERPC Overview

DCERPC is a protocol widely used by Microsoft distributed client and server applications that allows software clients to execute programs on a server remotely.

This typically involves a client querying a server called the Endpoint Mapper listening on a well known port number for the dynamically allocated network information of a required service. The client then sets up a secondary connection to the server instance providing the service. The security appliance allows the appropriate port number and network address and also applies NAT, if needed, for the secondary connection.
DCERPC inspect maps inspect for native TCP communication between the EPM and client on well-known TCP port 135. Map and lookup operations of the EPM are supported for clients. Client and server can be located in any security zone. The embedded server IP address and Port number are received from the applicable EPM response messages. Since a client may attempt multiple connections to the server port returned by EPM, multiple use of pinholes are allowed, which have user configurable timeouts.

**Note**
DCERPC inspection only supports communication between the EPM and clients to open pinholes through the ASASM. Clients using RPC communication that does not use the EPM is not supported with DCERPC inspection.

### Configuring a DCERPC Inspection Policy Map for Additional Inspection Control

To specify additional DCERPC inspection parameters, create a DCERPC inspection policy map. You can then apply the inspection policy map when you enable DCERPC inspection.

To create a DCERPC inspection policy map, perform the following steps:

**Step 1**
Create a DCERPC inspection policy map, enter the following command:

```
hostname(config)# policy-map type inspect dcerpc policy_map_name
```

Where the `policy_map_name` is the name of the policy map. The CLI enters policy-map configuration mode.

**Step 2**
(Optional) To add a description to the policy map, enter the following command:

```
hostname(config-pmap)# description string
```

**Step 3**
To configure parameters that affect the inspection engine, perform the following steps:

a. To enter parameters configuration mode, enter the following command:

```
hostname(config-pmap)# parameters
```

b. To configure the timeout for DCERPC pinholes and override the global system pinhole timeout of two minutes, enter the following command:

```
hostname(config-pmap-p)# timeout pinhole hh:mm:ss
```

Where the `hh:mm:ss` argument is the timeout for pinhole connections. Value is between 0:0:1 and 1193:0:0.

c. To configure options for the endpoint mapper traffic, enter the following command:

```
hostname(config-pmap-p)# endpoint-mapper [epm-service-only] [lookup-operation [timeout hh:mm:ss]]
```

Where the `hh:mm:ss` argument is the timeout for pinholes generated from the lookup operation. If no timeout is configured for the lookup operation, the timeout pinhole command or the default is used. The `epm-service-only` keyword enforces endpoint mapper service during binding so that only its service traffic is processed. The `lookup-operation` keyword enables the lookup operation of the endpoint mapper service.
The following example shows how to define a DCERPC inspection policy map with the timeout configured for DCERPC pinholes.

```
hostname(config)# policy-map type inspect dcerpc dcerpc_map
hostname(config-pmap)# timeout pinhole 0:10:00
hostname(config)# class-map dcerpc
hostname(config-cmap)# match port tcp eq 135
hostname(config)# policy-map global-policy
hostname(config-pmap)# class dcerpc
hostname(config-pmap-c)# inspect dcerpc dcerpc-map
hostname(config)# service-policy global-policy global
```

## GTP Inspection

This section describes the GTP inspection engine. This section includes the following topics:

- GTP Inspection Overview, page 43-3
- Configuring a GTP Inspection Policy Map for Additional Inspection Control, page 43-4
- Verifying and Monitoring GTP Inspection, page 43-8

---

**Note**

GTP inspection requires a special license. If you enter GTP-related commands on a ASASM without the required license, the ASASM displays an error message.

---

## GTP Inspection Overview

GPRS provides uninterrupted connectivity for mobile subscribers between GSM networks and corporate networks or the Internet. The GGSN is the interface between the GPRS wireless data network and other networks. The SGSN performs mobility, data session management, and data compression (See Figure 43-1).
The UMTS is the commercial convergence of fixed-line telephony, mobile, Internet and computer technology. UTRAN is the networking protocol used for implementing wireless networks in this system. GTP allows multi-protocol packets to be tunneled through a UMTS/GPRS backbone between a GGSN, an SGSN and the UTRAN.

GTP does not include any inherent security or encryption of user data, but using GTP with the ASASM helps protect your network against these risks.

The SGSN is logically connected to a GGSN using GTP. GTP allows multiprotocol packets to be tunneled through the GPRS backbone between GSNs. GTP provides a tunnel control and management protocol that allows the SGSN to provide GPRS network access for a mobile station by creating, modifying, and deleting tunnels. GTP uses a tunneling mechanism to provide a service for carrying user data packets.

When using GTP with failover, if a GTP connection is established and the active unit fails before data is transmitted over the tunnel, the GTP data connection (with a “j” flag set) is not replicated to the standby unit. This occurs because the active unit does not replicate embryonic connections to the standby unit.

### Configuring a GTP Inspection Policy Map for Additional Inspection Control

If you want to enforce additional parameters on GTP traffic, create and configure a GTP map. If you do not specify a map with the `inspect gtp` command, the ASASM uses the default GTP map, which is preconfigured with the following default values:

- `request-queue 200`
- `timeout gsn 0:30:00`
- `timeout pdp-context 0:30:00`
- `timeout request 0:01:00`
• timeout signaling 0:30:00
• timeout tunnel 0:01:00
• tunnel-limit 500

To create and configure a GTP map, perform the following steps. You can then apply the GTP map when you enable GTP inspection according to the “Configuring Application Layer Protocol Inspection” section on page 39-6.

Step 1 Create a GTP inspection policy map, enter the following command:

```
hostname(config)# policy-map type inspect gtp policy_map_name
```

Where the `policy_map_name` is the name of the policy map. The CLI enters policy-map configuration mode.

Step 2 (Optional) To add a description to the policy map, enter the following command:

```
hostname(config-pmap)# description string
```

Step 3 To match an Access Point name, enter the following command:

```
hostname(config-pmap)# match [not] apn regex [regex_name | class regex_class_name]
```

Step 4 To match a message ID, enter the following command:

```
hostname(config-pmap)# match [not] message id [message_id | range lower_range upper_range]
```

Where the `message_id` is an alphanumeric identifier between 1 and 255. The `lower_range` is lower range of message IDs. The `upper_range` is the upper range of message IDs.

Step 5 To match a message length, enter the following command:

```
hostname(config-pmap)# match [not] message length min min_length max max_length
```

Where the `min_length` and `max_length` are both between 1 and 65536. The length specified by this command is the sum of the GTP header and the rest of the message, which is the payload of the UDP packet.

Step 6 To match the version, enter the following command:

```
hostname(config-pmap)# match [not] version [version_id | range lower_range upper_range]
```

Where the `version_id` is between 0 and 255. The `lower_range` is lower range of versions. The `upper_range` is the upper range of versions.

Step 7 To configure parameters that affect the inspection engine, perform the following steps:

a. To enter parameters configuration mode, enter the following command:

```
hostname(config-pmap)# parameters
hostname(config-pmap-p)#
```

The `mnc network_code` argument is a two or three-digit value identifying the network code.

By default, the security appliance does not check for valid MCC/MNC combinations. This command is used for IMSI Prefix filtering. The MCC and MNC in the IMSI of the received packet is compared with the MCC/MNC configured with this command and is dropped if it does not match.
This command must be used to enable IMSI Prefix filtering. You can configure multiple instances to specify permitted MCC and MNC combinations. By default, the ASASM does not check the validity of MNC and MCC combinations, so you must verify the validity of the combinations configured. To find more information about MCC and MNC codes, see the ITU E.212 recommendation, Identification Plan for Land Mobile Stations.

b. To allow invalid GTP packets or packets that otherwise would fail parsing and be dropped, enter the following command:

```plaintext
hostname(config-pmap-p)# permit errors
```

By default, all invalid packets or packets that failed, during parsing, are dropped.

c. To enable support for GSN pooling, use the `permit response` command.

If the ASASM performs GTP inspection, by default the ASASM drops GTP responses from GSNs that were not specified in the GTP request. This situation occurs when you use load-balancing among a pool of GSNs to provide efficiency and scalability of GPRS.

You can enable support for GSN pooling by using the `permit response` command. This command configures the ASASM to allow responses from any of a designated set of GSNs, regardless of the GSN to which a GTP request was sent. You identify the pool of load-balancing GSNs as a network object. Likewise, you identify the SGSN as a network object. If the GSN responding belongs to the same object group as the GSN that the GTP request was sent to and if the SGSN is in an object group that the responding GSN is permitted to send a GTP response to, the ASASM permits the response.

d. To create an object to represent the pool of load-balancing GSNs, perform the following steps:

Use the `object-group` command to define a new network object group representing the pool of load-balancing GSNs.

```plaintext
hostname(config)# object-group network GSN-pool-name
hostname(config-network)#
```

For example, the following command creates an object group named gsnpool32:

```plaintext
hostname(config)# object-group network gsnpool32
hostname(config-network)#
```

e. Use the `network-object` command to specify the load-balancing GSNs. You can do so with one `network-object` command per GSN, using the `host` keyword. You can also use the `network-object` command to identify whole networks containing GSNs that perform load balancing.

```plaintext
hostname(config-network)# network-object host IP-address
```

For example, the following commands create three network objects representing individual hosts:

```plaintext
hostname(config-network)# network-object host 192.168.100.1
hostname(config-network)# network-object host 192.168.100.2
hostname(config-network)# network-object host 192.168.100.3
hostname(config-network)#
```

f. To create an object to represent the SGSN that the load-balancing GSNs are permitted to respond to, perform the following steps:

a. Use the `object-group` command to define a new network object group that will represent the SGSN that sends GTP requests to the GSN pool.

```plaintext
hostname(config)# object-group network SGSN-name
hostname(config-network)#
```

For example, the following command creates an object group named sgsn32:

```plaintext
hostname(config)# object-group network sgsn32
```
b. Use the `network-object` command with the `host` keyword to identify the SGSN.

```
hostname(config-network)# network-object host IP-address
```

For example, the following command creates a network objects representing the SGSN:

```
hostname(config-network)# network-object host 192.168.50.100
hostname(config-network)#
```

g. To allow GTP responses from any GSN in the network object representing the GSN pool, defined in c., d, to the network object representing the SGSN, defined in c., f., enter the following commands:

```
hostname(config)# gtp-map map_name
hostname(config-gtp-map)# permit response to-object-group SGSN-name from-object-group GSN-pool-name
```

For example, the following command permits GTP responses from any host in the object group named gsnpool32 to the host in the object group named sgsn32:

```
hostname(config-gtp-map)# permit response to-object-group sgsn32 from-object-group gsnpool32
```

The following example shows how to support GSN pooling by defining network objects for the GSN pool and the SGSN. An entire Class C network is defined as the GSN pool but you can identify multiple individual IP addresses, one per `network-object` command, instead of identifying whole networks. The example then modifies a GTP map to permit responses from the GSN pool to the SGSN.

```
hostname(config)# object-group network gsnpool32
hostname(config-network)# network-object 192.168.100.0 255.255.255.0
hostname(config)# object-group network sgsn32
hostname(config-network)# network-object host 192.168.50.100
hostname(config)# gtp-map gtp-policy
hostname(config-gtp-map)# permit response to-object-group sgsn32 from-object-group gsnpool32
```

h. To specify the maximum number of GTP requests that will be queued waiting for a response, enter the following command:

```
hostname(config-gtp-map)# request-queue max_requests
```

where the `max_requests` argument sets the maximum number of GTP requests that will be queued waiting for a response, from 1 to 4294967295. The default is 200.

When the limit has been reached and a new request arrives, the request that has been in the queue for the longest time is removed. The Error Indication, the Version Not Supported and the SGSN Context Acknowledge messages are not considered as requests and do not enter the request queue to wait for a response.

i. To change the inactivity timers for a GTP session, enter the following command:

```
hostname(config-gtp-map)# timeout {gsn | pdp-context | request | signaling | tunnel} hh:mm:ss
```

Enter this command separately for each timeout.

The `gsn` keyword specifies the period of inactivity after which a GSN will be removed.

The `pdp-context` keyword specifies the maximum period of time allowed before beginning to receive the PDP context.
The **request** keyword specifies the maximum period of time allowed before beginning to receive the GTP message.

The **signaling** keyword specifies the period of inactivity after which the GTP signaling will be removed.

The **tunnel** keyword specifies the period of inactivity after which the GTP tunnel will be torn down.

The **hh:mm:ss** argument is the timeout where **hh** specifies the hour, **mm** specifies the minutes, and **ss** specifies the seconds. The value **0** means never tear down.

j. To specify the maximum number of GTP tunnels allowed to be active on the ASASM, enter the following command:

```
hostname(config-gtp-map)# tunnel-limit max_tunnels
```

where the **max_tunnels** argument is the maximum number of tunnels allowed, from 1 to 4294967295. The default is 500.

New requests will be dropped once the number of tunnels specified by this command is reached.

The following example shows how to limit the number of tunnels in the network:

```
hostname(config)# policy-map type inspect gtp gmap
hostname(config-pmap)# parameters
hostname(config-pmap-p)# tunnel-limit 3000

hostname(config)# policy-map global_policy
hostname(config-pmap)# class inspection_default
hostname(config-pmap-c)# inspect gtp gmap

hostname(config)# service-policy global_policy global
```

### Verifying and Monitoring GTP Inspection

To display GTP configuration, enter the `show service-policy inspect gtp` command in privileged EXEC mode. For the detailed syntax for this command, see the command page in the command reference.

Use the `show service-policy inspect gtp statistics` command to show the statistics for GTP inspection. The following is sample output from the `show service-policy inspect gtp statistics` command:

```
hostname# show service-policy inspect gtp statistics
GPRS GTP Statistics:
    version_not_support        0   msg_too_short            0
    unknown_msg                 0   unexpected_sig_msg      0
    unexpected_data_msg        0   ie_duplicated            0
    mandatory_ie_missing       0   mandatory_ie_incorrect  0
    optional_ie_incorrect      0   ie_unknown               0
    ie_out_of_order            0   ie_unexpected            0
    total_forwarded            0   total_dropped            0
    signalling_msg_dropped     0   data_msg_dropped         0
    signalling_msg_forwarded   0   data_msg_forwarded       0
    total_created_pdp          0   total_deleted_pdp       0
    total_created_pdpmc       0   total_deleted_pdpmc     0
    pdp_non_existent           0
```

You can use the vertical bar (|) to filter the display. Type `?l` for more display filtering options.

The following is sample GSN output from the `show service-policy inspect gtp statistics gsn` command:
hostname\# show service-policy inspect gtp statistics gsn 9.9.9.9
1 in use, 1 most used, timeout 0:00:00

GTP GSN Statistics for 9.9.9.9, Idle 0:00:00, restart counter 0
- Tunnels Active 0
- Tunnels Created 0
- Tunnels Destroyed 0
- Total Messages Received 2
- Signaling Messages Data Messages
  - total received 2 0
  - dropped 0 0
  - forwarded 2 0

Use the `show service-policy inspect gtp pdp-context` command to display PDP context-related information. The following is sample output from the `show service-policy inspect gtp pdp-context` command:

hostname\# show service-policy inspect gtp pdp-context detail
1 in use, 1 most used, timeout 0:00:00

<table>
<thead>
<tr>
<th>Version</th>
<th>TID</th>
<th>MS Addr</th>
<th>SGSN Addr</th>
<th>Idle</th>
<th>APN</th>
</tr>
</thead>
<tbody>
<tr>
<td>v1</td>
<td>1234567890123425</td>
<td>10.0.1.1</td>
<td>10.0.0.2</td>
<td>0:00:13</td>
<td>gprs.cisco.com</td>
</tr>
</tbody>
</table>

user_name (IMSI): 214365870921435  
primary pdp: Y  
sgsn_addr_signal: 10.0.0.2  
ggsn_addr_signal: 10.1.1.1  
sgsn_addr_data: 10.0.0.2  
ggsn_addr_data: 10.1.1.1  
sgsn_control_teid: 0x0000001d1  
ggsn_control_teid: 0x6306ffa0  
seq_tpdo_up: 0  
seq_tpdo_down: 0  
signal_sequence: 0  
upstream_signal_flow: 0  
downstream_signal_flow: 0  
RAupdate_flow: 0

The PDP context is identified by the tunnel ID, which is a combination of the values for IMSI and NSAPI. A GTP tunnel is defined by two associated PDP contexts in different GSN nodes and is identified with a Tunnel ID. A GTP tunnel is necessary to forward packets between an external packet data network and a MS user.

You can use the vertical bar (|) to filter the display, as in the following example:

hostname\# show service-policy gtp statistics | grep gsn

---

**RADIUS Accounting Inspection**

This section describes the IM inspection engine. This section includes the following topics:

- RADIUS Accounting Inspection Overview, page 43-9
- Configuring a RADIUS Inspection Policy Map for Additional Inspection Control, page 43-10

---

**RADIUS Accounting Inspection Overview**

One of the well known problems is the over-billing attack in GPRS networks. The over-billing attack can cause consumers anger and frustration by being billed for services that they have not used. In this case, a malicious attacker sets up a connection to a server and obtains an IP address from the SGSN. When the attacker ends the call, the malicious server will still send packets to it, which gets dropped by
the GGSN, but the connection from the server remains active. The IP address assigned to the malicious attacker gets released and reassigned to a legitimate user who will then get billed for services that the attacker will use.

RADIUS accounting inspection prevents this type of attack by ensuring the traffic seen by the GGSN is legitimate. With the RADIUS accounting feature properly configured, the security appliance tears down a connection based on matching the Framed IP attribute in the Radius Accounting Request Start message with the Radius Accounting Request Stop message. When the Stop message is seen with the matching IP address in the Framed IP attribute, the security appliance looks for all connections with the source matching the IP address.

You have the option to configure a secret pre-shared key with the RADIUS server so the security appliance can validate the message. If the shared secret is not configured, the security appliance does not need to validate the source of the message and will only check that the source IP address is one of the configured addresses allowed to send the RADIUS messages.

**Note**

When using RADIUS accounting inspection with GPRS enabled, the ASASM checks for the 3GPP-Session-Stop-Indicator in the Accounting Request STOP messages to properly handle secondary PDP contexts. Specifically, the ASASM requires that the Accounting Request STOP messages include the 3GPP-SGSN-Address attribute before it will terminate the user sessions and all associated connections. Some third-party GGSNs might not send this attribute by default.

---

### Configuring a RADIUS Inspection Policy Map for Additional Inspection Control

In order to use this feature, the `radius-accounting-map` will need to be specified in the `policy-map type management` and then applied to the service-policy using the new `control-plane` keyword to specify that this traffic is for to-the-box inspection.

The following example shows the complete set of commands in context to properly configure this feature:

**Step 1**
Configure the class map and the port:

```
class-map type management c1
  match port udp eq 1888
```

**Step 2**
Create the policy map, and configure the parameters for RADIUS accounting inspection using the parameter command to access the proper mode to configure the attributes, host, and key.

```
policy-map type inspect radius-accounting radius_accounting_map
  parameters
    host 10.1.1.1 inside key 123456789
    send response
    enable gprs
    validate-attribute 22
```

**Step 3**
Configure the service policy and control-plane keywords.

```
policy-map type management global_policy
  class c1
    inspect radius-accounting radius_accounting_map

service-policy global_policy control-plane abc global
```
RSH Inspection

RSH inspection is enabled by default. The RSH protocol uses a TCP connection from the RSH client to the RSH server on TCP port 514. The client and server negotiate the TCP port number where the client listens for the STDERR output stream. RSH inspection supports NAT of the negotiated port number if necessary.

SNMP Inspection

This section describes the IM inspection engine. This section includes the following topics:

- SNMP Inspection Overview, page 43-11
- Configuring an SNMP Inspection Policy Map for Additional Inspection Control, page 43-11

SNMP Inspection Overview

SNMP application inspection lets you restrict SNMP traffic to a specific version of SNMP. Earlier versions of SNMP are less secure; therefore, denying certain SNMP versions may be required by your security policy. The ASASM can deny SNMP versions 1, 2, 2c, or 3. You control the versions permitted by creating an SNMP map.

You then apply the SNMP map when you enable SNMP inspection according to the “Configuring Application Layer Protocol Inspection” section on page 39-6.

Configuring an SNMP Inspection Policy Map for Additional Inspection Control

To create an SNMP inspection policy map, perform the following steps:

Step 1
To create an SNMP map, enter the following command:

```
hostname(config)# snmp-map map_name
hostname(config-snmp-map)#
```

where `map_name` is the name of the SNMP map. The CLI enters SNMP map configuration mode.

Step 2
To specify the versions of SNMP to deny, enter the following command for each version:

```
hostname(config-snmp-map)# deny version version
hostname(config-snmp-map)#
```

where `version` is 1, 2, 2c, or 3.

The following example denies SNMP Versions 1 and 2:

```
hostname(config)# snmp-map sample_map
hostname(config-snmp-map)# deny version 1
hostname(config-snmp-map)# deny version 2
```
XDMCP Inspection

XDMCP inspection is enabled by default; however, the XDMCP inspection engine is dependent upon proper configuration of the `established` command.

XDMCP is a protocol that uses UDP port 177 to negotiate X sessions, which use TCP when established. For successful negotiation and start of an XWindows session, the ASASM must allow the TCP back connection from the Xhosted computer. To permit the back connection, use the `established` command on the ASASM. Once XDMCP negotiates the port to send the display, the `established` command is consulted to verify if this back connection should be permitted.

During the XWindows session, the manager talks to the display Xserver on the well-known port 6000. Each display has a separate connection to the Xserver, as a result of the following terminal setting.

```shell
setenv DISPLAY Xserver:n
```

where `n` is the display number.

When XDMCP is used, the display is negotiated using IP addresses, which the ASASM can NAT if needed. XDCMP inspection does not support PAT.
PART 11

Configuring Connection Settings and QoS
Configuring Connection Settings

This chapter describes how to configure connection settings for connections that go through the ASASM, or for management connections, that go to the ASASM. Connection settings include:

- Maximum connections (TCP and UDP connections, embryonic connections, per-client connections)
- Connection timeouts
- Dead connection detection
- TCP sequence randomization
- TCP normalization customization
- TCP state bypass
- Global timeouts

This chapter includes the following sections:

- Information About Connection Settings, page 44-1
- Licensing Requirements for Connection Settings, page 44-4
- Guidelines and Limitations, page 44-5
- Default Settings, page 44-5
- Configuring Connection Settings, page 44-6
- Monitoring Connection Settings, page 44-14
- Configuration Examples for Connection Settings, page 44-14
- Feature History for Connection Settings, page 44-16

Information About Connection Settings

This section describes why you might want to limit connections and includes the following topics:

- TCP Intercept and Limiting Embryonic Connections, page 44-2
- Disabling TCP Intercept for Management Packets for Clientless SSL Compatibility, page 44-2
- Dead Connection Detection (DCD), page 44-2
- TCP Sequence Randomization, page 44-3
- TCP Normalization, page 44-3
- TCP State Bypass, page 44-3
TCP Intercept and Limiting Embryonic Connections

Limiting the number of embryonic connections protects you from a DoS attack. The ASASM uses the per-client limits and the embryonic connection limit to trigger TCP Intercept, which protects inside systems from a DoS attack perpetrated by flooding an interface with TCP SYN packets. An embryonic connection is a connection request that has not finished the necessary handshake between source and destination. TCP Intercept uses the SYN cookies algorithm to prevent TCP SYN-flooding attacks. A SYN-flooding attack consists of a series of SYN packets usually originating from spoofed IP addresses. The constant flood of SYN packets keeps the server SYN queue full, which prevents it from servicing connection requests. When the embryonic connection threshold of a connection is crossed, the ASASM acts as a proxy for the server and generates a SYN-ACK response to the client SYN request. When the ASASM receives an ACK back from the client, it can then authenticate the client and allow the connection to the server.

**Note**
When you use TCP SYN cookie protection to protect servers from SYN attacks, you must set the embryonic connection limit lower than the TCP SYN backlog queue on the server that you want to protect. Otherwise, valid clients can no longer access the server during a SYN attack.

To view TCP Intercept statistics, including the top 10 servers under attack, see Chapter 47, “Configuring Threat Detection.”

Disabling TCP Intercept for Management Packets for Clientless SSL Compatibility

By default, TCP management connections have TCP Intercept always enabled. When TCP Intercept is enabled, it intercepts the 3-way TCP connection establishment handshake packets and thus deprives the ASASM from processing the packets for clientless SSL. Clientless SSL requires the ability to process the 3-way handshake packets to provide selective ACK and other TCP options for clientless SSL connections. To disable TCP Intercept for management traffic, you can set the embryonic connection limit; only after the embryonic connection limit is reached is TCP Intercept enabled.

Dead Connection Detection (DCD)

DCD detects a dead connection and allows it to expire, without expiring connections that can still handle traffic. You configure DCD when you want idle, but valid connections to persist.

When you enable DCD, idle timeout behavior changes. With idle timeout, DCD probes are sent to each of the two end-hosts to determine the validity of the connection. If an end-host fails to respond after probes are sent at the configured intervals, the connection is freed, and reset values, if configured, are sent to each of the end-hosts. If both end-hosts respond that the connection is valid, the activity timeout is updated to the current time and the idle timeout is rescheduled accordingly.

Enabling DCD changes the behavior of idle-timeout handling in the TCP normalizer. DCD probing resets the idle timeout on the connections seen in the `show conn` command. To determine when a connection that has exceeded the configured timeout value in the timeout command but is kept alive due to DCD probing, the `show service-policy` command includes counters to show the amount of activity from DCD.
TCP Sequence Randomization

Each TCP connection has two ISNs: one generated by the client and one generated by the server. The ASASM randomizes the ISN of the TCP SYN passing in both the inbound and outbound directions.

Randomizing the ISN of the protected host prevents an attacker from predicting the next ISN for a new connection and potentially hijacking the new session.

TCP initial sequence number randomization can be disabled if required. For example:

- If another in-line firewall is also randomizing the initial sequence numbers, there is no need for both firewalls to be performing this action, even though this action does not affect the traffic.
- If you use eBGP multi-hop through the ASASM, and the eBGP peers are using MD5. Randomization breaks the MD5 checksum.
- You use a WAAS device that requires the ASASM not to randomize the sequence numbers of connections.

TCP Normalization

The TCP normalization feature identifies abnormal packets that the ASASM can act on when they are detected; for example, the ASASM can allow, drop, or clear the packets. TCP normalization helps protect the ASASM from attacks. TCP normalization is always enabled, but you can customize how some features behave.

The TCP normalizer includes non-configurable actions and configurable actions. Typically, non-configurable actions that drop or clear connections apply to packets that are always bad. Configurable actions (as detailed in “Customizing the TCP Normalizer with a TCP Map” section on page 44-6) might need to be customized depending on your network needs.

See the following guidelines for TCP normalization:

- The normalizer does not protect from SYN floods. The ASASM includes SYN flood protection in other ways.
- The normalizer always sees the SYN packet as the first packet in a flow unless the ASASM is in loose mode due to failover.

TCP State Bypass

By default, all traffic that goes through the ASASM is inspected using the Adaptive Security Algorithm and is either allowed through or dropped based on the security policy. The ASASM maximizes the firewall performance by checking the state of each packet (is this a new connection or an established
connection?) and assigning it to either the session management path (a new connection SYN packet), the fast path (an established connection), or the control plane path (advanced inspection). See the “Stateful Inspection Overview” section on page 1-10 for more detailed information about the stateful firewall.

TCP packets that match existing connections in the fast path can pass through the ASASM without rechecking every aspect of the security policy. This feature maximizes performance. However, the method of establishing the session in the fast path using the SYN packet, and the checks that occur in the fast path (such as TCP sequence number), can stand in the way of asymmetrical routing solutions: both the outbound and inbound flow of a connection must pass through the same ASASM.

For example, a new connection goes to ASASM 1. The SYN packet goes through the session management path, and an entry for the connection is added to the fast path table. If subsequent packets of this connection go through ASASM 1, then the packets will match the entry in the fast path, and are passed through. But if subsequent packets go to ASASM 2, where there was not a SYN packet that went through the session management path, then there is no entry in the fast path for the connection, and the packets are dropped. Figure 44-1 shows an asymmetric routing example where the outbound traffic goes through a different ASASM than the inbound traffic:

![Asymmetric Routing Diagram](image)

If you have asymmetric routing configured on upstream routers, and traffic alternates between two ASASMs, then you can configure TCP state bypass for specific traffic. TCP state bypass alters the way sessions are established in the fast path and disables the fast path checks. This feature treats TCP traffic much as it treats a UDP connection: when a non-SYN packet matching the specified networks enters the ASASM, and there is not an fast path entry, then the packet goes through the session management path to establish the connection in the fast path. Once in the fast path, the traffic bypasses the fast path checks.

### Licensing Requirements for Connection Settings

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>
Guidelines and Limitations

This section includes the following guidelines and limitations:

- TCP State Bypass Guidelines and Limitations, page 44-5

TCP State Bypass Guidelines and Limitations

Context Mode Guidelines
Supported in single and multiple context mode.

Firewall Mode Guidelines
Supported in routed and transparent mode.

Failover Guidelines
Failover is supported.

Unsupported Features
The following features are not supported when you use TCP state bypass:

- Application inspection—Application inspection requires both inbound and outbound traffic to go through the same ASASM, so application inspection is not supported with TCP state bypass.
- AAA authenticated sessions—When a user authenticates with one ASASM, traffic returning via the other ASASM will be denied because the user did not authenticate with that ASASM.
- TCP Intercept, maximum embryonic connection limit, TCP sequence number randomization—The ASASM does not keep track of the state of the connection, so these features are not applied.
- TCP normalization—The TCP normalizer is disabled.
- SSM and SSC functionality—You cannot use TCP state bypass and any application running on an SSM or SSC, such as IPS or CSC.

NAT Guidelines
Because the translation session is established separately for each ASASM, be sure to configure static NAT on both ASASMs for TCP state bypass traffic; if you use dynamic NAT, the address chosen for the session on ASASM 1 will differ from the address chosen for the session on ASASM 2.

Default Settings

TCP State Bypass
TCP state bypass is disabled by default.

TCP Normalizer
The default configuration includes the following settings:

- no check-retransmission
- no checksum-verification
Configuring Connection Settings

This section includes the following topics:

- Customizing the TCP Normalizer with a TCP Map, page 44-6
- Configuring Connection Settings, page 44-10

Task Flow For Configuring Configuration Settings (Except Global Timeouts)

**Step 1**
For TCP normalization customization, create a TCP map according to the “Customizing the TCP Normalizer with a TCP Map” section on page 44-6.

**Step 2**
For all connection settings except for global timeouts, configure a service policy according to Chapter 30, “Configuring a Service Policy Using the Modular Policy Framework.”

**Step 3**
Configure connection settings according to the “Configuring Connection Settings” section on page 44-10.

Customizing the TCP Normalizer with a TCP Map

To customize the TCP normalizer, first define the settings using a TCP map.

Detailed Steps

**Step 1**
To specify the TCP normalization criteria that you want to look for, create a TCP map by entering the following command:

```
hostname(config)# tcp-map tcp-map-name
```

For each TCP map, you can customize one or more settings.

**Step 2**
(Optional) Configure the TCP map criteria by entering one or more of the following commands (see Table 44-1). If you want to customize some settings, then the defaults are used for any commands you do not enter.

```text
exceed-mss allow
queue-limit 0 timeout 4
reserved-bits allow
syn-data allow
synack-data drop
invalid-ack drop
seq-past-window drop
tcp-options range 6 7 clear
tcp-options range 9 255 clear
tcp-options selective-ack allow
tcp-options timestamp allow
tcp-options window-scale allow
ttl-evasion-protection
urgent-flag clear
window-variation allow-connection
```
### Table 44-1 tcp-map Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>check-retransmission</td>
<td>Prevents inconsistent TCP retransmissions.</td>
</tr>
<tr>
<td>checksum-verification</td>
<td>Verifies the checksum.</td>
</tr>
</tbody>
</table>
| exceed-mss {allow | drop}  | Sets the action for packets whose data length exceeds the TCP maximum segment size.  
  (Default) The **allow** keyword allows packets whose data length exceeds the TCP maximum segment size.  
  The **drop** keyword drops packets whose data length exceeds the TCP maximum segment size. |
| invalid-ack {allow | drop}  | Sets the action for packets with an invalid ACK. You might see invalid ACKs in the following instances:  
  • In the TCP connection SYN-ACK-received status, if the ACK number of a received TCP packet is not exactly same as the sequence number of the next TCP packet sending out, it is an invalid ACK.  
  • Whenever the ACK number of a received TCP packet is greater than the sequence number of the next TCP packet sending out, it is an invalid ACK.  
  The **allow** keyword allows packets with an invalid ACK.  
  (Default) The **drop** keyword drops packets with an invalid ACK.  
  **Note** TCP packets with an invalid ACK are automatically allowed for WAAS connections. |
**Configuring Connection Settings**

**queue-limit pkt_num [timeout seconds]**

Sets the maximum number of out-of-order packets that can be buffered and put in order for a TCP connection, between 1 and 250 packets. The default is 0, which means this setting is disabled and the default system queue limit is used depending on the type of traffic:

- Connections for application inspection (the `inspect` command), IPS (the `ips` command), and TCP check-retransmission (the TCP map `check-retransmission` command) have a queue limit of 3 packets. If the ASASM receives a TCP packet with a different window size, then the queue limit is dynamically changed to match the advertised setting.

- For other TCP connections, out-of-order packets are passed through untouched.

If you set the `queue-limit` command to be 1 or above, then the number of out-of-order packets allowed for all TCP traffic matches this setting. For example, for application inspection, IPS, and TCP check-retransmission traffic, any advertised settings from TCP packets are ignored in favor of the `queue-limit` setting. For other TCP traffic, out-of-order packets are now buffered and put in order instead of passed through untouched.

The `timeout seconds` argument sets the maximum amount of time that out-of-order packets can remain in the buffer, between 1 and 20 seconds; if they are not put in order and passed on within the timeout period, then they are dropped. The default is 4 seconds. You cannot change the timeout for any traffic if the `pkt_num` argument is set to 0; you need to set the limit to be 1 or above for the `timeout` keyword to take effect.

**reserved-bits {allow | clear | drop}**

Sets the action for reserved bits in the TCP header.

(_Default) The `allow` keyword allows packets with the reserved bits in the TCP header.

The `clear` keyword clears the reserved bits in the TCP header and allows the packet.

The `drop` keyword drops the packet with the reserved bits in the TCP header.

**seq-past-window {allow | drop}**

Sets the action for packets that have past-window sequence numbers, namely the sequence number of a received TCP packet is greater than the right edge of the TCP receiving window.

The `allow` keyword allows packets that have past-window sequence numbers. This action is only allowed if the `queue-limit` command is set to 0 (disabled).

(_Default) The `drop` keyword drops packets that have past-window sequence numbers.
### Table 44-1 tcp-map Commands (continued)

<table>
<thead>
<tr>
<th>Command</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>synack-data {allow</td>
<td>drop}</td>
</tr>
<tr>
<td>syn-data {allow</td>
<td>drop}</td>
</tr>
<tr>
<td>tcp-options {selective-ack</td>
<td>timestamp</td>
</tr>
<tr>
<td>tcp-options range lower upper {allow</td>
<td>clear</td>
</tr>
<tr>
<td>ttl-evasion-protection</td>
<td>Enables the TTL evasion protection. Do not disable this command if you want to prevent attacks that attempt to evade security policy. For example, an attacker can send a packet that passes policy with a very short TTL. When the TTL goes to zero, a router between the ASASM and the endpoint drops the packet. It is at this point that the attacker can send a malicious packet with a long TTL that appears to the ASASM to be a retransmission and is passed. To the endpoint host, however, it is the first packet that has been received by the attacker. In this case, an attacker is able to succeed without security preventing the attack.</td>
</tr>
</tbody>
</table>
### Configuring Connection Settings

To set connection settings, perform the following steps.

**Guidelines and Limitations**

Depending on the number of CPU cores on your ASASM model, the maximum concurrent and embryonic connections may exceed the configured numbers due to the way each core manages connections. In the worst case scenario, the ASASM allows up to $n-1$ extra connections and embryonic connections, where $n$ is the number of cores. For example, if your model has 4 cores, if you configure 6 concurrent connections and 4 embryonic connections, you could have an additional 3 of each type. To determine the number of cores for your model, enter the `show cpu core` command.

### Table 44-1  tcp-map Commands (continued)

<table>
<thead>
<tr>
<th>Command</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>`urgent-flag {allow</td>
<td>clear}`</td>
</tr>
<tr>
<td>`window-variation {allow</td>
<td>drop}`</td>
</tr>
</tbody>
</table>
### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> class-map name</td>
<td>Creates a class map to identify the traffic for which you want to disable stateful firewall inspection.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# class-map bypass_traffic</td>
</tr>
<tr>
<td><strong>Step 2</strong> match parameter</td>
<td>Specifies the traffic in the class map. See the “Identifying Traffic (Layer 3/4 Class Maps)” section on page 30-12 for more information.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config-cmap)# match access-list bypass</td>
</tr>
<tr>
<td><strong>Step 3</strong> policy-map name</td>
<td>Adds or edits a policy map that sets the actions to take with the class map traffic.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# policy-map tcp_bypass_policy</td>
</tr>
<tr>
<td><strong>Step 4</strong> class name</td>
<td>Identifies the class map created in Step 1</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config-pmap)# class bypass_traffic</td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td>Do one or more of the following:</td>
</tr>
</tbody>
</table>
## Configuring Connection Settings

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`set connection {[conn-max n] [embryonic-conn-max n] [per-client-embryonic-max n] [per-client-max n] [random-sequence-number {enable</td>
<td>disable}]}`</td>
</tr>
</tbody>
</table>

**Example:**
```
hostname(config-pmap-c)# set connection
conn-max 256 random-sequence-number
disable
```

### Note
For management traffic, you can only set the `conn-max` and `embryonic-conn-max` keywords.
Chapter 44        Configuring Connection Settings

Configuring Connection Settings

**set connection timeout** {embryonic \( hh:mm:ss \)} [idle \( hh:mm:ss \) [reset]] [half-closed \( hh:mm:ss \)] [dcd \( hh:mm:ss \) [max_retries]]

**Example:**

```
hostname(config-pmap-c)# set connection timeout idle 2:0:0 embryonic 0:40:0 half-closed 0:20:0 dcd
```

**Purpose**

Sets connection timeouts.

The **embryonic** \( hh:mm:ss \) keyword sets the timeout period until a TCP embryonic (half-open) connection is closed, between 0:5 and 1193:0:0. The default is 0:0:30. You can also set this value to 0, which means the connection never times out.

The **idle** \( hh:mm:ss \) keyword sets the idle timeout for all protocols between 0:0:1 and 1193:0:0. The default is 1:0:0. You can also set this value to 0, which means the connection never times out. For TCP traffic, the **reset** keyword sends a reset to TCP endpoints when the connection times out.

The **half-closed** \( hh:mm:ss \) keyword sets the idle timeout between 0:5:0 and 1193:0:0. The default is 0:10:0. Half-closed connections are not affected by DCD. Also, the ASASM does not send a reset when taking down half-closed connections.

The **dcd** keyword enables DCD. DCD detects a dead connection and allows it to expire, without expiring connections that can still handle traffic. You configure DCD when you want idle, but valid connections to persist. After a TCP connection times out, the ASASM sends DCD probes to the end hosts to determine the validity of the connection. If one of the end hosts fails to respond after the maximum retries are exhausted, the ASASM frees the connection. If both end hosts respond that the connection is valid, the ASASM updates the activity timeout to the current time and reschedules the idle timeout accordingly. The **retry-interval** sets the time duration in \( hh:mm:ss \) format to wait after each unresponsive DCD probe before sending another probe, between 0:0:1 and 24:0:0. The default is 0:0:15. The **max-retries** sets the number of consecutive failed retries for DCD before declaring the connection as dead. The minimum value is 1 and the maximum value is 255. The default is 5.

The default **tcp** idle timeout is 1 hour.

The default **udp** idle timeout is 2 minutes.

The default **icmp** idle timeout is 2 seconds.

The default **esp** and **ha** idle timeout is 30 seconds.

For all other protocols, the default idle timeout is 2 minutes.

To never time out, enter 0:0:0.

You can enter this command all on one line (in any order), or you can enter each attribute as a separate command. The command is combined onto one line in the running configuration.

**Note**

This command is not available for management traffic.
Monitoring Connection Settings

This section includes the following topics:

- Monitoring TCP State Bypass, page 44-14

Monitoring TCP State Bypass

To monitor TCP state bypass, perform one of the following tasks:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show conn</td>
<td>If you use the show conn command, the display for connections that use TCP state bypass includes the flag “b.”</td>
</tr>
</tbody>
</table>

Configuration Examples for Connection Settings

This section includes the following topics:

- Configuration Examples for Connection Limits and Timeouts, page 44-15
- Configuration Examples for TCP State Bypass, page 44-15
- Configuration Examples for TCP Normalization, page 44-15
Configuration Examples for Connection Limits and Timeouts

The following example sets the connection limits and timeouts for all traffic:

```
hostname(config)# class-map CONNS
hostname(config-cmap)# match any
hostname(config-cmap)# policy-map CONNS
hostname(config-pmap-c)# class CONNS
hostname(config-pmap-c)# set connection conn-max 1000 embryonic-conn-max 3000
hostname(config-pmap-c)# set connection timeout idle 2:0:0 embryonic 0:40:0 half-closed 0:20:0 dcd
hostname(config-pmap-c)# service-policy CONNS interface outside
```

You can enter `set connection` commands with multiple parameters or you can enter each parameter as a separate command. The ASASM combines the commands into one line in the running configuration. For example, if you entered the following two commands in class configuration mode:

```
hostname(config-pmap-c)# set connection conn-max 600
hostname(config-pmap-c)# set connection embryonic-conn-max 50
```

the output of the `show running-config policy-map` command would display the result of the two commands in a single, combined command:

```
set connection conn-max 600 embryonic-conn-max 50
```

Configuration Examples for TCP State Bypass

The following is a sample configuration for TCP state bypass:

```
hostname(config)# access-list tcp_bypass extended permit tcp 10.1.1.0 255.255.255.224 any
hostname(config)# class-map tcp_bypass
hostname(config-cmap)# description "TCP traffic that bypasses stateful firewall"
hostname(config-cmap)# match access-list tcp_bypass
hostname(config-cmap)# policy-map tcp_bypass_policy
hostname(config-pmap)# class tcp_bypass
hostname(config-pmap-c)# set connection advanced-options tcp-state-bypass
hostname(config-pmap-c)# service-policy tcp_bypass_policy outside
hostname(config-pmap-c)# static (inside,outside) 209.165.200.224 10.1.1.0 netmask 255.255.255.224
```

Configuration Examples for TCP Normalization

For example, to allow urgent flag and urgent offset packets for all traffic sent to the range of TCP ports between the well known FTP data port and the Telnet port, enter the following commands:

```
hostname(config)# tcp-map tmap
hostname(config-tcp-map)# urgent-flag allow
hostname(config-tcp-map)# class-map urg-class
hostname(config-cmap)# match port tcp range ftp-data telnet
hostname(config-cmap)# policy-map pmap
hostname(config-pmap)# class urg-class
hostname(config-pmap-c)# set connection advanced-options tmap
hostname(config-pmap-c)# service-policy pmap global
```
Feature History for Connection Settings

Table 44-2 lists each feature change and the platform release in which it was implemented.

Table 44-2  Feature History for Connection Settings

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCP state bypass</td>
<td>8.2(1)</td>
<td>This feature was introduced. The following command was introduced: <code>set connection advanced-options tcp-state-bypass</code>.</td>
</tr>
</tbody>
</table>
| Connection timeout for all protocols | 8.2(2)           | The idle timeout was changed to apply to all protocols, not just TCP.  
The following command was modified: `set connection timeout` |
Configuring QoS

Have you ever participated in a long-distance phone call that involved a satellite connection? The conversation might be interrupted with brief, but perceptible, gaps at odd intervals. Those gaps are the time, called the latency, between the arrival of packets being transmitted over the network. Some network traffic, such as voice and video, cannot tolerate long latency times. Quality of service (QoS) is a feature that lets you give priority to critical traffic, prevent bandwidth hogging, and manage network bottlenecks to prevent packet drops.

Note

For the ASASM, we suggest performing QoS on the switch instead of the ASASM. Switches have more capability in this area.

This chapter describes how to apply QoS policies and includes the following sections:

- Information About QoS, page 45-1
- Licensing Requirements for QoS, page 45-3
- Guidelines and Limitations, page 45-3
- Configuring QoS, page 45-3
- Monitoring QoS, page 45-6
- Feature History for QoS, page 45-6

Information About QoS

You should consider that in an ever-changing network environment, QoS is not a one-time deployment, but an ongoing, essential part of network design.

This section describes the QoS features supported by the ASASM and includes the following topics:

- Supported QoS Features, page 45-2
- What is a Token Bucket?, page 45-2
- Information About Policing, page 45-2
- DSCP and DiffServ Preservation, page 45-2
Supported QoS Features

The ASASM supports the following QoS features:

- **Policing**—To prevent individual flows from hogging the network bandwidth, you can limit the maximum bandwidth used per flow. See the “Information About Policing” section on page 45-2 for more information.

What is a Token Bucket?

A token bucket is used to manage a device that regulates the data in a flow. For example, the regulator might be a traffic policer or a traffic shaper. A token bucket itself has no discard or priority policy. Rather, a token bucket discards tokens and leaves to the flow the problem of managing its transmission queue if the flow overdrives the regulator.

A token bucket is a formal definition of a rate of transfer. It has three components: a burst size, an average rate, and a time interval. Although the average rate is generally represented as bits per second, any two values may be derived from the third by the relation shown as follows:

average rate = burst size / time interval

Here are some definitions of these terms:

- **Average rate**—Also called the committed information rate (CIR), it specifies how much data can be sent or forwarded per unit time on average.

- **Burst size**—Also called the Committed Burst (Bc) size, it specifies in bits or bytes per burst how much traffic can be sent within a given unit of time to not create scheduling concerns. (For traffic shaping, it specifies bits per burst; for policing, it specifies bytes per burst.)

- **Time interval**—Also called the measurement interval, it specifies the time quantum in seconds per burst.

In the token bucket metaphor, tokens are put into the bucket at a certain rate. The bucket itself has a specified capacity. If the bucket fills to capacity, newly arriving tokens are discarded. Each token is permission for the source to send a certain number of bits into the network. To send a packet, the regulator must remove from the bucket a number of tokens equal in representation to the packet size.

If not enough tokens are in the bucket to send a packet, the packet either waits until the bucket has enough tokens (in the case of traffic shaping) or the packet is discarded or marked down (in the case of policing). If the bucket is already full of tokens, incoming tokens overflow and are not available to future packets. Thus, at any time, the largest burst a source can send into the network is roughly proportional to the size of the bucket.

Information About Policing

Policing is a way of ensuring that no traffic exceeds the maximum rate (in bits/second) that you configure, thus ensuring that no one traffic flow or class can take over the entire resource. When traffic exceeds the maximum rate, the ASASM drops the excess traffic. Policing also sets the largest single burst of traffic allowed.

DSCP and DiffServ Preservation

- DSCP markings are preserved on all traffic passing through the ASASM.
• The ASASM does not locally mark/remark any classified traffic, but it honors the Expedited Forwarding (EF) DSCP bits of every packet to determine if it requires “priority” handling and will direct those packets to the LLQ.
• DiffServ marking is preserved on packets when they traverse the service provider backbone so that QoS can be applied in transit (QoS tunnel pre-classification).

Licensing Requirements for QoS

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

**Context Mode Guidelines**
Supported in single context mode only. Does not support multiple context mode.

**Firewall Mode Guidelines**
Supported in routed firewall mode only. Does not support transparent firewall mode.

**IPv6 Guidelines**
Does not support IPv6.

**Model Guidelines**
• (ASASM) Only policing is supported.

Configuring QoS

This section includes the following topics:
• Configuring a Service Rule for Policing, page 45-3

Configuring a Service Rule for Policing

To create a policy map, perform the following steps.

**Restrictions**
• You cannot use the class-default class map for priority traffic.
Guidelines

- For policing traffic, you can choose to police all other traffic, or you can limit the traffic to certain types.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>class-map policing_map_name</td>
</tr>
</tbody>
</table>

Example:
hostname(config)# class-map policing_traffic

For policing traffic, creates a class map to identify the traffic for which you want to perform policing.

| **Step 2** | match parameter |

Example:
hostname(config-cmap)# match access-list policing

Specifies the traffic in the class map. See the “Identifying Traffic (Layer 3/4 Class Maps)” section on page 30-12 for more information.

| **Step 3** | policy-map name |

Example:
hostname(config)# policy-map QoS_policy

Adds or edits a policy map.

| **Step 4** | class policing_map_name |

Example:
hostname(config-pmap)# class policing_class

Identifies the class map you created for policed traffic in Step 1.
### Configuring QoS

#### Examples

**Example 45-1 Policing Example**

In this example, the maximum rate for traffic of the tcp_traffic class is 56,000 bits/second and a maximum burst size of 10,500 bytes per second. For the TG1-BestEffort class, the maximum rate is 200,000 bits/second, with a maximum burst of 37,500 bytes/second.

hostname(config)# access-list tcp_traffic permit tcp any any
hostname(config)# class-map tcp_traffic
hostname(config-cmap)# match access-list tcp_traffic

hostname(config-cmap)# class-map TG1-BestEffort
hostname(config-cmap)# match tunnel-group tunnel-grpl
hostname(config-cmap)# match flow ip destination-address

hostname(config)# policy-map qos
hostname(config-pmap)# class tcp_traffic
hostname(config-pmap-c)# police output 56000 10500

hostname(config-pmap-c)# class TG1-best-effort
hostname(config-pmap-c)# police output 200000 37500

hostname(config-pmap-c)# class class-default
hostname(config-pmap-c)# police output 1000000 37500

hostname(config-pmap-c)# service-policy qos global
Monitoring QoS

This section includes the following topics:

- Viewing QoS Police Statistics, page 45-6

Viewing QoS Police Statistics

To view the QoS statistics for traffic policing, use the `show service-policy` command with the `police` keyword:

```
hostname# show service-policy police
```

The following is sample output for the `show service-policy police` command:

```
hostname# show service-policy police
Global policy:
  Service-policy: global_fw_policy
Interface outside:
  Service-policy: qos
  Class-map: browse
    police Interface outside:
      cir 56000 bps, bc 10500 bytes
      conformed 10065 packets, 12621510 bytes; actions: transmit
      exceeded 499 packets, 625146 bytes; actions: drop
      conformed 5600 bps, exceed 5016 bps
  Class-map: cmap2
    police Interface outside:
      cir 200000 bps, bc 37500 bytes
      conformed 17179 packets, 20614800 bytes; actions: transmit
      exceeded 617 packets, 770718 bytes; actions: drop
      conformed 198785 bps, exceed 2303 bps
```

Feature History for QoS

Table 45-1 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority queuing and policing</td>
<td>7.0(1)</td>
<td>We introduced QoS priority queuing and policing. We introduced the following commands: priority-queue, queue-limit, tx-ring-limit, priority, police, show priority-queue statistics, show service-policy police, show service-policy priority, show running-config priority-queue, clear configure priority-queue</td>
</tr>
</tbody>
</table>
PART 12

Configuring Advanced Network Protection
Configuring the Botnet Traffic Filter

Malware is malicious software that is installed on an unknowing host. Malware that attempts network activity such as sending private data (passwords, credit card numbers, key strokes, or proprietary data) can be detected by the Botnet Traffic Filter when the malware starts a connection to a known bad IP address. The Botnet Traffic Filter checks incoming and outgoing connections against a dynamic database of known bad domain names and IP addresses (the blacklist), and then logs or blocks any suspicious activity.

You can also supplement the Cisco dynamic database with blacklisted addresses of your choosing by adding them to a static blacklist; if the dynamic database includes blacklisted addresses that you think should not be blacklisted, you can manually enter them into a static whitelist. Whitelisted addresses still generate syslog messages, but because you are only targeting blacklist syslog messages, they are informational.

Note

If you do not want to use the Cisco dynamic database at all, because of internal requirements, you can use the static blacklist alone if you can identify all the malware sites that you want to target.

This chapter describes how to configure the Botnet Traffic Filter and includes the following sections:

- Information About the Botnet Traffic Filter, page 46-1
- Licensing Requirements for the Botnet Traffic Filter, page 46-6
- Guidelines and Limitations, page 46-6
- Default Settings, page 46-6
- Configuring the Botnet Traffic Filter, page 46-6
- Monitoring the Botnet Traffic Filter, page 46-17
- Configuration Examples for the Botnet Traffic Filter, page 46-19
- Where to Go Next, page 46-21
- Feature History for the Botnet Traffic Filter, page 46-22

Information About the Botnet Traffic Filter

This section includes information about the Botnet Traffic Filter and includes the following topics:

- Botnet Traffic Filter Address Types, page 46-2
- Botnet Traffic Filter Actions for Known Addresses, page 46-2
Botnet Traffic Filter Address Types

Addresses monitored by the Botnet Traffic Filter include:

- Known malware addresses—These addresses are on the blacklist identified by the dynamic database and the static blacklist.
- Known allowed addresses—These addresses are on the whitelist. The whitelist is useful when an address is blacklisted by the dynamic database and also identified by the static whitelist.
- Ambiguous addresses—These addresses are associated with multiple domain names, but not all of these domain names are on the blacklist. These addresses are on the greylist.
- Unlisted addresses—These addresses are unknown, and not included on any list.

Botnet Traffic Filter Actions for Known Addresses

You can configure the Botnet Traffic Filter to log suspicious activity, and you can optionally configure it to block suspicious traffic automatically.

Unlisted addresses do not generate any syslog messages, but addresses on the blacklist, whitelist, and greylist generate syslog messages differentiated by type. See the “Botnet Traffic Filter Syslog Messaging” section on page 46-17 for more information.

Botnet Traffic Filter Databases

The Botnet Traffic Filter uses two databases for known addresses. You can use both databases together, or you can disable use of the dynamic database and use the static database alone. This section includes the following topics:

- Information About the Dynamic Database, page 46-2
- Information About the Static Database, page 46-3
- Information About the DNS Reverse Lookup Cache and DNS Host Cache, page 46-4

Information About the Dynamic Database

The Botnet Traffic Filter can receive periodic updates for the dynamic database from the Cisco update server. This database lists thousands of known bad domain names and IP addresses.

How the ASA Uses the Dynamic Database

The ASASM uses the dynamic database as follows:

1. When the domain name in a DNS reply matches a name in the dynamic database, the Botnet Traffic Filter adds the name and IP address to the DNS reverse lookup cache.
2. When the infected host starts a connection to the IP address of the malware site, then the ASASM sends a syslog message informing you of the suspicious activity and optionally drops the traffic if you configured the ASASM to do so.
3. In some cases, the IP address itself is supplied in the dynamic database, and the Botnet Traffic Filter logs or drops any traffic to that IP address without having to inspect DNS requests.

Database Files

The database files are stored in running memory; they are not stored in flash memory. If you need to delete the database, use the `dynamic-filter database purge` command instead. Be sure to first disable use of the database by entering the `no dynamic-filter use-database` command.

**Note**

To use the database, be sure to configure a domain name server for the ASASM so that it can access the URL.

To use the domain names in the dynamic database, you need to enable DNS packet inspection with Botnet Traffic Filter snooping; the ASASM looks inside the DNS packets for the domain name and associated IP address.

Database Traffic Types

The dynamic database includes the following types of addresses:

- **Ads**—Advertising networks that deliver banner ads, interstitials, rich media ads, pop-ups, and pop-unders for websites, spyware and adware. Some of these networks send ad-oriented HTML emails and email verification services.
- **Data Tracking**—Sources associated with companies and websites that offer data tracking and metrics services to websites and other online entities. Some of these also run small advertising networks.
- **Spyware**—Sources that distribute spyware, adware, greyware, and other potentially unwanted advertising software. Some of these also run exploits to install such software.
- **Malware (higher threat level)**—Sources that use various exploits to deliver adware, spyware and other malware to victim computers. Some of these are associated with rogue online vendors and distributors of dialers which deceptively call premium-rate phone numbers.
- **Malware (lower threat level)**—Sources that deliver deceptive or malicious anti-spyware, anti-malware, registry cleaning, and system cleaning software.
- **Adult**—Sources associated with adult networks/services offering web hosting for adult content, advertising, content aggregation, registration and billing, and age verification. These may be tied to distribution of adware, spyware, and dialers.
- **Bot and Threat Networks**—Rogue systems that control infected computers. They are either systems hosted on threat networks or systems that are part of the botnet itself.
  - (Conficker) Bot and Threat Networks—Command-and-control servers or botnet-masters of Conficker botnets.
  - (ZeusBotnet) Bot and Threat Networks—Command-and-control servers or botnet-masters of Zeus botnets.

Information About the Static Database

You can manually enter domain names or IP addresses (host or subnet) that you want to tag as bad names in a blacklist. Static blacklist entries are always designated with a Very High threat level. You can also enter names or IP addresses in a whitelist, so that names or addresses that appear on both the `dynamic`
blacklist and the whitelist are identified only as whitelist addresses in syslog messages and reports. Note that you see syslog messages for whitelisted addresses even if the address is not also in the dynamic blacklist.

When you add a domain name to the static database, the ASASM waits 1 minute, and then sends a DNS request for that domain name and adds the domain name/IP address pairing to the DNS host cache. (This action is a background process, and does not affect your ability to continue configuring the ASASM). We recommend also enabling DNS packet inspection with Botnet Traffic Filter snooping. The ASASM uses Botnet Traffic Filter snooping instead of the regular DNS lookup to resolve static blacklist domain names in the following circumstances:

- The ASASM DNS server is unavailable.
- A connection is initiated during the 1 minute waiting period before the ASASM sends the regular DNS request.

If DNS snooping is used, when an infected host sends a DNS request for a name on the static database, the ASASM looks inside the DNS packets for the domain name and associated IP address and adds the name and IP address to the DNS reverse lookup cache.

If you do not enable Botnet Traffic Filter snooping, and one of the above circumstances occurs, then that traffic will not be monitored by the Botnet Traffic Filter.

**Information About the DNS Reverse Lookup Cache and DNS Host Cache**

When you use the dynamic database with DNS snooping, entries are added to the DNS reverse lookup cache. If you use the static database, entries are added to the DNS host cache (see the “Information About the Static Database” section on page 46-3 about using the static database with DNS snooping and the DNS reverse lookup cache).

Entries in the DNS reverse lookup cache and the DNS host cache have a time to live (TTL) value provided by the DNS server. The largest TTL value allowed is 1 day (24 hours); if the DNS server provides a larger TTL, it is truncated to 1 day maximum.

For the DNS reverse lookup cache, after an entry times out, the ASASM renews the entry when an infected host initiates a connection to a known address, and DNS snooping occurs.

For the DNS host cache, after an entry times out, the ASASM periodically requests a refresh for the entry.

For the DNS host cache, the maximum number of blacklist entries and whitelist entries is 1000 each.

**Table 46-1 DNS Reverse Lookup Cache Entries per Model**

<table>
<thead>
<tr>
<th>ASA Model</th>
<th>Maximum Entries</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASA 5505</td>
<td>5000</td>
</tr>
<tr>
<td>ASA 5510</td>
<td>10,000</td>
</tr>
<tr>
<td>ASA 5520</td>
<td>20,000</td>
</tr>
<tr>
<td>ASA 5540</td>
<td>40,000</td>
</tr>
<tr>
<td>ASA 5550</td>
<td>40,000</td>
</tr>
<tr>
<td>ASA 5580</td>
<td>100,000</td>
</tr>
</tbody>
</table>
How the Botnet Traffic Filter Works

Figure 46-1 shows how the Botnet Traffic Filter works with the dynamic database plus DNS inspection with Botnet Traffic Filter snooping.

Figure 46-1  How the Botnet Traffic Filter Works with the Dynamic Database

Figure 46-2 shows how the Botnet Traffic Filter works with the static database.

Figure 46-2  How the Botnet Traffic Filter Works with the Static Database
Licensing Requirements for the Botnet Traffic Filter

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>You need the following licenses:</td>
</tr>
<tr>
<td></td>
<td>• Botnet Traffic Filter License.</td>
</tr>
<tr>
<td></td>
<td>• Strong Encryption (3DES/AES) License to download the dynamic database.</td>
</tr>
</tbody>
</table>

Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

Context Mode Guidelines
Supported in single and multiple context mode.

Firewall Mode Guidelines
Supported in routed and transparent firewall mode.

Failover Guidelines
Does not support replication of the DNS reverse lookup cache, DNS host cache, or the dynamic database in Stateful Failover.

IPv6 Guidelines
Does not support IPv6.

Additional Guidelines and Limitations
- TCP DNS traffic is not supported.
- You can add up to 1000 blacklist entries and 1000 whitelist entries in the static database.

Default Settings

By default, the Botnet Traffic Filter is disabled, as is use of the dynamic database.
For DNS inspection, which is enabled by default, Botnet Traffic Filter snooping is disabled by default.

Configuring the Botnet Traffic Filter

This section includes the following topics:
- Task Flow for Configuring the Botnet Traffic Filter, page 46-7
- Configuring the Dynamic Database, page 46-7
Task Flow for Configuring the Botnet Traffic Filter

To configure the Botnet Traffic Filter, perform the following steps:

**Step 1**
Enable use of the dynamic database. See the “Configuring the Dynamic Database” section on page 46-7.
This procedure enables database updates from the Cisco update server, and also enables use of the downloaded dynamic database by the ASASM. Disallowing use of the downloaded database is useful in multiple context mode so you can configure use of the database on a per-context basis.

**Step 2**
(Optional) Add static entries to the database. See the “Adding Entries to the Static Database” section on page 46-9.
This procedure lets you augment the dynamic database with domain names or IP addresses that you want to blacklist or whitelist. You might want to use the static database instead of the dynamic database if you do not want to download the dynamic database over the Internet.

**Step 3**
Enable DNS snooping. See the “Enabling DNS Snooping” section on page 46-10.
This procedure enables inspection of DNS packets, compares the domain name with those in the dynamic database or the static database (when a DNS server for the ASASM is unavailable), and adds the name and IP address to the DNS reverse lookup cache. This cache is then used by the Botnet Traffic Filter when connections are made to the suspicious address.

**Step 4**
Enable traffic classification and actions for the Botnet Traffic Filter. See the “Enabling Traffic Classification and Actions for the Botnet Traffic Filter” section on page 46-12.
This procedure enables the Botnet Traffic Filter, which compares the source and destination IP address in each initial connection packet to the IP addresses in the dynamic database, static database, DNS reverse lookup cache, and DNS host cache, and sends a syslog message or drops any matching traffic.

**Step 5**
(Optional) Block traffic manually based on syslog message information. See the “Blocking Botnet Traffic Manually” section on page 46-15.
If you choose not to block malware traffic automatically, you can block traffic manually by configuring an access list to deny traffic, or by using the `shun` command to block all traffic to and from a host.

Configuring the Dynamic Database

This procedure enables database updates, and also enables use of the downloaded dynamic database by the ASASM. Disabling use of the downloaded database is useful in multiple context mode so you can configure use of the database on a per-context basis.

By default, downloading and using the dynamic database is disabled.
Prerequisites

Enable ASASM use of a DNS server according to the “Configuring the DNS Server” section on page 9-8.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><strong>dynamic-filter updater-client enable</strong></td>
</tr>
</tbody>
</table>

**Example:**
```
hostname(config)# dynamic-filter updater-client enable
```

| **Step 2** | *(Multiple context mode only)* | changeto context *context_name* | Changes to the context so that you can configure use of the database on a per-context basis. |

**Example:**
```
hostname# changeto context admin
hostname/admin#
```

| **Step 3** | **dynamic-filter use-database** | Enables use of the dynamic database. In multiple context mode, enter this command in the context execution space. |

**Example:**
```
hostname(config)# dynamic-filter use-database
```

Examples

The following multiple mode example enables downloading of the dynamic database, and enables use of the database in context1 and context2:

```
hostname(config)# dynamic-filter updater-client enable
hostname(config)# changeto context context1
hostname/context1(config)# dynamic-filter use-database
hostname/context1(config)# changeto context context2
hostname/context2(config)# dynamic-filter use-database
```

The following single mode example enables downloading of the dynamic database, and enables use of the database:

```
hostname(config)# dynamic-filter updater-client enable
hostname(config)# dynamic-filter use-database
```

What to Do Next

See the “Adding Entries to the Static Database” section on page 46-9.
Adding Entries to the Static Database

The static database lets you augment the dynamic database with domain names or IP addresses that you want to blacklist or whitelist. Static blacklist entries are always designated with a Very High threat level. See the “Information About the Static Database” section on page 46-3 for more information.

Prerequisites

- In multiple context mode, perform this procedure in the context execution space.
- Enable ASASM use of a DNS server according to the “Configuring the DNS Server” section on page 9-8.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td><code>dynamic-filter blacklist</code></td>
<td>Edits the Botnet Traffic Filter blacklist.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# dynamic-filter blacklist</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td>Enter one or both of the following:</td>
<td></td>
</tr>
<tr>
<td><code>name domain_name</code></td>
<td>Adds a name to the blacklist. You can enter this command multiple times for multiple entries. You can add up to 1000 blacklist entries.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config-llist)# name bad.example.com</td>
<td></td>
</tr>
<tr>
<td><code>address ip_address mask</code></td>
<td>Adds an IP address to the blacklist. You can enter this command multiple times for multiple entries. The <em>mask</em> can be for a single host or for a subnet.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config-llist)# address 10.1.1.1 255.255.255.255</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td></td>
</tr>
<tr>
<td><code>dynamic-filter whitelist</code></td>
<td>Edits the Botnet Traffic Filter whitelist.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# dynamic-filter whitelist</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td></td>
</tr>
<tr>
<td>Enter one or both of the following:</td>
<td></td>
</tr>
</tbody>
</table>
Configuring the Botnet Traffic Filter

Examples

The following example creates entries for the blacklist and whitelist:

```
hostname(config)# dynamic-filter blacklist
hostname(config-llist)# name bad1.example.com
hostname(config-llist)# name bad2.example.com
hostname(config-llist)# address 10.1.1.1 255.255.255.0
hostname(config-llist)# dynamic-filter whitelist
hostname(config-llist)# name good.example.com
hostname(config-llist)# name great.example.com
hostname(config-llist)# name awesome.example.com
hostname(config-llist)# address 10.1.1.2 255.255.255.255
```

What to Do Next

See the “Enabling DNS Snooping” section on page 46-10.

Enabling DNS Snooping

This procedure enables inspection of DNS packets and enables Botnet Traffic Filter snooping, which compares the domain name with those on the dynamic database or static database, and adds the name and IP address to the Botnet Traffic Filter DNS reverse lookup cache. This cache is then used by the Botnet Traffic Filter when connections are made to the suspicious address.

The following procedure creates an interface-specific service policy for DNS inspection. See the “DNS Inspection” section on page 40-1 and Chapter 30, “Configuring a Service Policy Using the Modular Policy Framework,” for detailed information about configuring advanced DNS inspection options using the Modular Policy Framework.

Prerequisites

In multiple context mode, perform this procedure in the context execution space.

Restrictions

TCP DNS traffic is not supported.
Default DNS Inspection Configuration and Recommended Configuration

The default configuration for DNS inspection inspects all UDP DNS traffic on all interfaces, and does not have DNS snooping enabled.

We suggest that you enable DNS snooping only on interfaces where external DNS requests are going. Enabling DNS snooping on all UDP DNS traffic, including that going to an internal DNS server, creates unnecessary load on the ASASM.

For example, if the DNS server is on the outside interface, you should enable DNS inspection with snooping for all UDP DNS traffic on the outside interface. See the “Examples” section for the recommended commands for this configuration.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td>class-map name</td>
<td>Creates a class map to identify the traffic for which you want to inspect DNS.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# class-map dynamic-filter_snoop_class</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td>match parameters</td>
<td>Specifies traffic for the class map. See the “Identifying Traffic (Layer 3/4 Class Maps)” section on page 30-12 for more information about available parameters. For example, you can specify an access list for DNS traffic to and from certain addresses, or you can specify all UDP DNS traffic.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config-cmap)# match port udp eq domain</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td></td>
</tr>
<tr>
<td>policy-map name</td>
<td>Adds or edits a policy map so you can set the actions to take with the class map traffic.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# policy-map dynamic-filter_snoop_policy</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td></td>
</tr>
<tr>
<td>class name</td>
<td>Identifies the class map you created in <strong>Step 1</strong>.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config-pmap)# class dynamic-filter_snoop_class</td>
<td></td>
</tr>
</tbody>
</table>
Chapter 46      Configuring the Botnet Traffic Filter

Configuring the Botnet Traffic Filter

Examples

The following recommended configuration creates a class map for all UDP DNS traffic, enables DNS inspection and Botnet Traffic Filter snooping with the default DNS inspection policy map, and applies it to the outside interface:

```
hostname(config)# class-map dynamic-filter_snoop_class
hostname(config-cmap)# match port udp eq domain
hostname(config-cmap)# policy-map dynamic-filter_snoop_policy
hostname(config-pmap-c)# inspect dns preset_dns_map dynamic-filter-snoop
hostname(config-pmap-c)# service-policy dynamic-filter_snoop_policy interface outside
```

What to Do Next

See the “Enabling Traffic Classification and Actions for the Botnet Traffic Filter” section on page 46-12.

Enabling Traffic Classification and Actions for the Botnet Traffic Filter

This procedure enables the Botnet Traffic Filter. The Botnet Traffic Filter compares the source and destination IP address in each initial connection packet to the following:

- Dynamic database IP addresses
- Static database IP addresses
- DNS reverse lookup cache (for dynamic database domain names)
- DNS host cache (for static database domain names)

When an address matches, the ASASM sends a syslog message. The only additional action currently available is to drop the connection.

Prerequisites

In multiple context mode, perform this procedure in the context execution space.

---

### Command Table

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td><code>inspect dns [map_name]</code></td>
<td>Enables DNS inspection with Botnet Traffic Filter snooping. To use the default DNS inspection policy map for the <code>map_name</code>, specify <code>preset_dns_map</code> for the map name. See the “DNS Inspection” section on page 40-1 for more information about creating a DNS inspection policy map.</td>
</tr>
<tr>
<td></td>
<td><code>dynamic-filter-snoop</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>hostname(config-pmap-c)# inspect dns</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>preset_dns_map dynamic-filter-snoop</code></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td><code>service-policy policymap_name interface</code></td>
<td>Activates the policy map on an interface. The interface-specific policy overrides the global policy. You can only apply one policy map to each interface.</td>
</tr>
<tr>
<td></td>
<td><code>interface_name</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>hostname(config)# service-policy</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>dynamic-filter_snoop_policy interface</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>outside</code></td>
<td></td>
</tr>
</tbody>
</table>
Recommended Configuration

Although DNS snooping is not required, we recommend configuring DNS snooping for maximum use of the Botnet Traffic Filter (see the “Enabling DNS Snooping” section on page 46-10). Without DNS snooping for the dynamic database, the Botnet Traffic Filter uses only the static database entries, plus any IP addresses in the dynamic database; domain names in the dynamic database are not used.

We recommend enabling the Botnet Traffic Filter on all traffic on the Internet-facing interface, and enabling dropping of traffic with a severity of moderate and higher. See the “Examples” section for the recommended commands used for this configuration.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> (Optional)</td>
<td>Identifies the traffic that you want to monitor or drop. If you do not create an access list for monitoring, by default you monitor all traffic. You can optionally use an access list to identify a subset of monitored traffic that you want to drop; be sure the access list is a subset of the monitoring access list. See Chapter 14, “Adding an Extended Access List,” for more information about creating an access list.</td>
</tr>
<tr>
<td>`access-list access_list_name extended (deny</td>
<td>permit) protocol source_address mask [operator port] dest_address mask [operator port]`</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# access-list dynamic-filter_acl extended permit tcp any any eq 80</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# access-list dynamic-filter_acl_subset extended permit tcp 10.1.1.0 255.255.255.0 any eq 80</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Enables the Botnet Traffic Filter; without any options, this command monitors all traffic.</td>
</tr>
<tr>
<td><code>dynamic-filter enable [interface name] [classify-list access_list]</code></td>
<td>You can optionally limit monitoring to specific traffic by using the <code>classify-list</code> keyword with an access list.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>You can enter this command one time for each interface and one time for the global policy (where you do not specify the <code>interface</code> keyword). Each interface and global command can have an optional <code>classify-list</code> keyword. Any interface-specific commands take precedence over the global command.</td>
</tr>
<tr>
<td>hostname(config)# dynamic-filter enable interface outside classify-list dynamic-filter_acl</td>
<td></td>
</tr>
</tbody>
</table>
## Step 3

(Optional)

**Command**

```
dynamic-filter drop blacklist [interface name] [action-classify-list subset_access_list] [threat-level {eq level | range min max}]
```

### Example:

```
hostname(config)# dynamic-filter drop blacklist interface outside action-classify-list
dynamic-filter_acl_subset threat-level range moderate very-high
```

### Purpose

Automatically drops malware traffic. To manually drop traffic, see the “Blocking Botnet Traffic Manually” section on page 46-15.

Be sure to first configure a `dynamic-filter enable` command to monitor any traffic you also want to drop.

You can set an interface policy using the `interface` keyword, or a global policy (where you do not specify the `interface` keyword). Any interface-specific commands take precedence over the global command. You can enter this command multiple times for each interface and global policy.

The `action-classify-list` keyword limits the traffic dropped to a subset of monitored traffic. The dropped traffic must always be equal to or a subset of the monitored traffic. For example, if you specify an access list for the `dynamic-filter enable` command, and you specify the `action-classify-list` for this command, then it must be a subset of the `dynamic-filter enable` access list.

Make sure you do not specify overlapping traffic in multiple commands for a given interface/global policy. Because you cannot control the exact order that commands are matched, overlapping traffic means you do not know which command will be matched. For example, do not specify both a command that matches all traffic (without the `action-classify-list` keyword) as well as a command with the `action-classify-list` keyword for a given interface. In this case, the traffic might never match the command with the `action-classify-list` keyword. Similarly, if you specify multiple commands with the `action-classify-list` keyword, make sure each access list is unique, and that the networks do not overlap.

You can additionally limit the traffic dropped by setting the threat level. If you do not explicitly set a threat level, the level used is `threat-level range moderate very-high`.

**Note** We highly recommend using the default setting unless you have strong reasons for changing the setting.

The `level` and `min` and `max` options are:

- very-low
- low
- moderate
- high
- very-high

**Note** Static blacklist entries are always designated with a Very High threat level.
### Examples

The following recommended configuration monitors all traffic on the outside interface and drops all traffic at a threat level of moderate or higher:

```
hostname(config)# dynamic-filter enable interface outside
hostname(config)# dynamic-filter drop blacklist interface outside
```

If you decide not to monitor all traffic, you can limit the traffic using an access list. The following example monitors only port 80 traffic on the outside interface, and drops traffic threat level very-high only:

```
hostname(config)# access-list dynamic-filter_acl extended permit tcp any any eq 80
hostname(config)# dynamic-filter enable interface outside classify-list dynamic-filter_acl
hostname(config)# dynamic-filter drop blacklist interface outside threat-level eq very-high
```

### Blocking Botnet Traffic Manually

If you choose not to block malware traffic automatically (see the “Enabling Traffic Classification and Actions for the Botnet Traffic Filter” section on page 46-12), you can block traffic manually by configuring an access list to deny traffic, or by using the `shun` command tool to block all traffic to and from a host.

For example, you receive the following syslog message:


You can then perform one of the following actions:

- Create an access list to deny traffic.

  For example, using the syslog message above, you might want to deny traffic from the infected host at 10.1.1.45 to the malware site at 209.165.202.129. Or, if there are many connections to different blacklisted addresses, you can create an access list to deny all traffic from 10.1.1.45 until you resolve the infection on the host computer. For example, the following commands deny all traffic from 10.1.1.5 to 209.165.202.129, but permits all other traffic on the inside interface:

  ```
  hostname(config)# access-list BLOCK_OUT extended deny ip host 10.1.1.45 host 209.165.202.129
  hostname(config)# access-list BLOCK_OUT extended permit ip any any
  hostname(config)# access-group BLOCK_OUT in interface inside
  ```

### Step 4

(Optional)

**Command**

```
dynamic-filter ambiguous-is-black
```

**Purpose**

If you configured the `dynamic-filter drop blacklist` command, then this command treats greylisted traffic as blacklisted traffic for dropping purposes. If you do not enable this command, greylisted traffic will not be dropped. See the “Botnet Traffic Filter Address Types” section on page 46-2 for more information about the greylist.

### Example:

```
hostname(config)# dynamic-filter ambiguous-is-black
```
See Chapter 14, “Adding an Extended Access List,” for more information about creating an access list, and see Chapter 32, “Configuring Access Rules,” for information about applying the access list to the interface.

**Note**
Access lists block all future connections. To block the current connection, if it is still active, enter the **clear conn** command. For example, to clear only the connection listed in the syslog message, enter the **clear conn address 10.1.1.45 address 209.165.202.129** command. See the command reference for more information.

- Shun the infected host.

  Shunning blocks all connections from the host, so you should use an access list if you want to block connections to certain destination addresses and ports. To shun a host, enter the following command. To drop the current connection as well as blocking all future connections, enter the destination address, source port, destination port, and optional protocol.
  
  \[
  \text{hostname(config)} \# \text{shun src_ip} [\text{dst_ip src_port dest_port} [\text{protocol}]]
  \]

  For example, to block future connections from 10.1.1.45, and also drop the current connection to the malware site in the syslog message, enter:
  
  \[
  \text{hostname(config)} \# \text{shun 10.1.1.45 209.165.202.129 6798 80}
  \]

  See “Blocking Unwanted Connections” section on page 48-2 for more information about shunning.

  After you resolve the infection, be sure to remove the access list or the shun. To remove the shun, enter **no shun src_ip**.

### Searching the Dynamic Database

If you want to check if a domain name or IP address is included in the dynamic database, you can search the database for a string.

#### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>dynamic-filter database find string</strong></td>
<td>Searches the dynamic database for a domain name or IP address. The string can be the complete domain name or IP address, or you can enter part of the name or address, with a minimum search string of 3 characters. If there are multiple matches, the first two matches are shown. To refine your search for a more specific match, enter a longer string.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname# <strong>dynamic-filter database find</strong></td>
<td></td>
</tr>
</tbody>
</table>

**Note** Regular expressions are not supported for the database search.

#### Examples

The following example searches on the string “example.com”, and finds 1 match:

```
hostname# **dynamic-filter database find** bad.example.com

bad.example.com
Found 1 matches
```

The following example searches on the string “bad”, and finds more than 2 matches:
hostname# dynamic-filter database find bad

    bad.example.com
    bad.example.net
Found more than 2 matches, enter a more specific string to find an exact match

Monitoring the Botnet Traffic Filter

Whenever a known address is classified by the Botnet Traffic Filter, then a syslog message is generated. You can also monitor Botnet Traffic Filter statistics and other parameters by entering commands on the ASASM. This section includes the following topics:

- Botnet Traffic Filter Syslog Messaging, page 46-17
- Botnet Traffic Filter Commands, page 46-17

Botnet Traffic Filter Syslog Messaging

The Botnet Traffic Filter generates detailed syslog messages numbered 338nnn. Messages differentiate between incoming and outgoing connections, blacklist, whitelist, or greylist addresses, and many other variables. (The greylist includes addresses that are associated with multiple domain names, but not all of these domain names are on the blacklist.)

See the syslog messages guide for detailed information about syslog messages.

Botnet Traffic Filter Commands

To monitor the Botnet Traffic Filter, enter one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show dynamic-filter statistics [interface name] [detail]</td>
<td>Shows how many connections were classified as whitelist, blacklist, and greylist connections, and how many connections were dropped. (The greylist includes addresses that are associated with multiple domain names, but not all of these domain names are on the blacklist.) The detail keyword shows how many packets at each threat level were classified or dropped. To clear the statistics, enter the clear dynamic-filter statistics [interface name] command.</td>
</tr>
<tr>
<td>show dynamic-filter reports top [malware-sites</td>
<td>malware-ports</td>
</tr>
</tbody>
</table>
Chapter 46  Configuring the Botnet Traffic Filter

Monitoring the Botnet Traffic Filter

Examples

The following is sample output from the `show dynamic-filter statistics` command:

```
hostname# show dynamic-filter statistics
Enabled on interface outside
Total conns classified 11, ingress 11, egress 0
Total whitelist classified 0, ingress 0, egress 0
Total greylist classified 0, dropped 0, ingress 0, egress 0
Total blacklist classified 11, dropped 5, ingress 11, egress 0
Enabled on interface inside
Total conns classified 1182, ingress 1182, egress 0
Total whitelist classified 3, ingress 3, egress 0
Total greylist classified 0, dropped 0, ingress 0, egress 0
Total blacklist classified 1179, dropped 1000, ingress 1179, egress 0
```

The following is sample output from the `show dynamic-filter reports top malware-sites` command:

```
hostname# show dynamic-filter reports top malware-sites
Site Connections logged dropped Threat Level Category
bad1.example.com (10.67.22.34) 11 0 2 Botnet
bad2.example.com (209.165.200.225) 8 8 3 Virus
bad1.cisco.example(10.131.36.158) 6 6 3 Virus
bad2.cisco.example(209.165.201.1) 2 2 3 Trojan
```

Command | Purpose
--- | ---
`show dynamic-filter reports infected-hosts {max-connections | latest-active | highest-threat | subnet ip_address netmask | all}` | Generates reports about infected hosts. These reports contain detailed history about infected hosts, showing the correlation between infected hosts, visited malware sites, and malware ports. The `max-connections` keyword shows the 20 infected hosts with the most number of connections. The `latest-active` keyword shows the 20 hosts with the most recent activity. The `highest-threat` keyword shows the 20 hosts that connected to the malware sites with the highest threat level. The `subnet` keyword shows up to 20 hosts within the specified subnet. The `all` keyword shows all buffered infected-hosts information. This display might include thousands of entries. You might want to use ASDM to generate a PDF file instead of using the CLI.

To clear the report data, enter the `clear dynamic-filter reports infected-hosts` command.

`show dynamic-filter updater-client` | Shows information about the updater server, including the server IP address, the next time the ASASM will connect with the server, and the database version last installed.

`show dynamic-filter dns-snoop [detail]` | Shows the Botnet Traffic Filter DNS snooping summary, or with the `detail` keyword, the actual IP addresses and names. All inspected DNS data is included in this output, and not just matching names in the blacklist. DNS data from static entries are not included.

To clear the DNS snooping data, enter the `clear dynamic-filter dns-snoop` command.

`show dynamic-filter data` | Shows information about the dynamic database, including when the dynamic database was last downloaded, the version of the database, how many entries the database contains, and 10 sample entries.

`show asp table dynamic-filter [hits]` | Shows the Botnet Traffic Filter rules that are installed in the accelerated security path.
Configuration Examples for the Botnet Traffic Filter

This section includes the recommended configuration for single and multiple context mode, as well as other possible configurations. This section includes the following topics:

- Recommended Configuration Example, page 46-19
- Other Configuration Examples, page 46-20

Recommended Configuration Example

The following recommended example configuration for single context mode enables downloading of the dynamic database, and enables use of the database. It creates a class map for all UDP DNS traffic, enables DNS inspection and Botnet Traffic Filter snooping with the default DNS inspection policy map, and applies it to the outside interface, the Internet-facing interface.

Example 46-1  Single Mode Botnet Traffic Filter Recommended Example

hostname(config)# dynamic-filter updater-client enable
hostname(config)# dynamic-filter use-database
hostname(config)# class-map dynamic-filter_snoop_class
hostname(config-cmap)# match port udp eq domain
hostname(config-cmap)# policy-map dynamic-filter_snoop_policy
hostname(config-pmap)# class dynamic-filter_snoop_class

horrible.example.net(10.232.224.2) 2 2 3 Botnet
nono.example.org(209.165.202.130) 1 1 3 Virus

Last clearing of the top sites report: at 13:41:06 UTC Jul 15 2009

The following is sample output from the show dynamic-filter reports top malware-ports command:

hostname# show dynamic-filter reports top malware-ports
Port                                      Connections logged
----------------------------------------------------------------------
tcp 1000                                      617
tcp 2001                                      472
tcp 23                                        22
tcp 1001                                      19
udp 2000                                     17
udp 2001                                     17
tcp 8080                                      9
tcp 80                                        3
tcp >8192                                     2

Last clearing of the top sites report: at 13:41:06 UTC Jul 15 2009

The following is sample output from the show dynamic-filter reports top infected-hosts command:

hostname# show dynamic-filter reports top infected-hosts
Host                                      Connections logged
----------------------------------------------------------------------
10.10.10.51(inside)                          1190
10.12.10.10(inside)                          10
10.10.11.10(inside)                         5

Last clearing of the top infected-hosts report: at 13:41:06 UTC Jul 15 2009
hostname(config-pmap-c)# inspect dns preset_dns_map dynamic-filter-snoop
hostname(config-pmap-c)# service-policy dynamic-filter_snoop_policy_interface outside
hostname(config)# dynamic-filter enable interface outside
hostname(config)# dynamic-filter drop blacklist interface outside

The following recommended example configuration for multiple context mode enables the Botnet Traffic Filter for two contexts:

**Example 46-2  Multiple Mode Botnet Traffic Filter Recommended Example**

hostname(config)# dynamic-filter updater-client enable
hostname(config)# changeto context context1
hostname/context1(config)# dynamic-filter use-database
hostname/context1(config)# class-map dynamic-filter_snoop_class
hostname/context1(config-cmap)# match port udp eq domain
hostname/context1(config-cmap)# policy-map dynamic-filter_snoop_policy
hostname/context1(config-pmap)# class dynamic-filter_snoop_class
hostname/context1(config-pmap-c)# inspect dns preset_dns_map dynamic-filter-snoop
hostname/context1(config-pmap-c)# service-policy dynamic-filter_snoop_policy_interface outside
hostname/context1(config)# dynamic-filter enable interface outside
hostname/context1(config)# dynamic-filter drop blacklist interface outside
hostname/context1(config)# changeto context context2
hostname/context2(config)# dynamic-filter use-database
hostname/context2(config)# class-map dynamic-filter_snoop_class
hostname/context2(config-cmap)# match port udp eq domain
hostname/context2(config-cmap)# policy-map dynamic-filter_snoop_policy
hostname/context2(config-pmap)# class dynamic-filter_snoop_class
hostname/context2(config-pmap-c)# inspect dns preset_dns_map dynamic-filter-snoop
hostname/context2(config-pmap-c)# service-policy dynamic-filter_snoop_policy_interface outside
hostname/context2(config)# dynamic-filter enable interface outside
hostname/context2(config)# dynamic-filter drop blacklist interface outside

**Other Configuration Examples**

The following sample configuration adds static entries are to the blacklist and to the whitelist. Then, it monitors all port 80 traffic on the outside interface, and drops blacklisted traffic. It also treats greylist addresses as blacklisted addresses.

hostname(config)# dynamic-filter updater-client enable
hostname(config)# changeto context context1
hostname/context1(config)# dynamic-filter use-database
hostname/context1(config)# class-map dynamic-filter_snoop_class
hostname/context1(config-cmap)# match port udp eq domain
hostname/context1(config-cmap)# policy-map dynamic-filter_snoop_policy
hostname/context1(config-pmap)# class dynamic-filter_snoop_class
hostname/context1(config-pmap-c)# inspect dns preset_dns_map dynamic-filter-snoop
hostname/context1(config-pmap-c)# service-policy dynamic-filter_snoop_policy_interface outside
hostname/context1(config-llist)# name bad1.example.com
hostname/context1(config-llist)# name bad2.example.com
hostname/context1(config-llist)# address 10.1.1.1 255.255.255.0
hostname/context1(config-llist)# dynamic-filter whitelist
hostname/context1(config-llist)# name good.example.com
hostname/context1(config-llist)# name great.example.com
hostname/context1(config-llist)# name awesome.example.com
hostname/context1(config-llist)# address 10.1.1.2 255.255.255.255
hostname/context1(config-llist)# access-list dynamic-filter_acl extended permit tcp any eq 80
hostname/context1(config)# dynamic-filter enable interface outside classify-list
dynamic-filter_acl
hostname/context1(config)# dynamic-filter drop blacklist interface outside
dynamic-filter_acl
hostname/context1(config)# dynamic-filter ambiguous-is-black
dynamic-filter_acl
hostname/context2(config)# changeto context context2
hostname/context2(config)# dynamic-filter use-database
dynamic-filter_acl
hostname/context2(config)# class-map dynamic-filter_snoop_class
dynamic-filter_acl
hostname/context2(config-cmap)# match port udp eq domain
dynamic-filter_acl
hostname/context2(config-cmap)# policy-map dynamic-filter_snoop_policy
dynamic-filter_acl
hostname/context2(config-pmap)# class dynamic-filter_snoop_class
dynamic-filter_acl
hostname/context2(config-pmap-c)# inspect dns preset_dns_map dynamic-filter-snoop
service-policy dynamic-filter_snoop_policy
outside
dynamic-filter_acl
hostname/context2(config-pmap-c)# dynamic-filter blacklist
dynamic-filter_acl
hostname/context2(config-llist)# name bad1.example.com
dynamic-filter_acl
hostname/context2(config-llist)# name bad2.example.com
dynamic-filter_acl
hostname/context2(config-llist)# address 10.1.1.1 255.255.255.0
dynamic-filter_acl
hostname/context2(config-llist)# dynamic-filter whitelist
dynamic-filter_acl
hostname/context2(config-llist)# name good.example.com
hostname/context2(config-llist)# name great.example.com
hostname/context2(config-llist)# name awesome.example.com
hostname/context2(config-llist)# address 10.1.1.2 255.255.255.255
dynamic-filter_acl
hostname/context2(config-llist)# access-list dynamic-filter_acl extended permit tcp any eq 80
hostname/context2(config)# dynamic-filter enable interface outside classify-list
dynamic-filter_acl
hostname/context2(config)# dynamic-filter drop blacklist interface outside
dynamic-filter_acl
hostname/context2(config)# dynamic-filter ambiguous-is-black
dynamic-filter_acl

Where to Go Next

- To configure the syslog server, see Chapter 52, “Configuring Logging.”
- To configure an access list to block traffic, see Chapter 14, “Adding an Extended Access List,” and also see Chapter 32, “Configuring Access Rules,” for information about applying the access list to the interface.
- To shun connections, see the “Blocking Unwanted Connections” section on page 48-2.
# Feature History for the Botnet Traffic Filter

Table 46-2 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Botnet Traffic Filter</td>
<td>8.2(1)</td>
<td>This feature was introduced.</td>
</tr>
<tr>
<td>Automatic blocking, and blacklist category and threat level reporting.</td>
<td>8.2(2)</td>
<td>The Botnet Traffic Filter now supports automatic blocking of blacklisted traffic based on the threat level. You can also view the category and threat level of malware sites in statistics and reports. The 1 hour timeout for reports for top hosts was removed; there is now no timeout. The following commands were introduced or modified: dynamic-filter ambiguous-is-black, dynamic-filter drop blacklist, show dynamic-filter statistics, show dynamic-filter reports infected-hosts, and show dynamic-filter reports top.</td>
</tr>
</tbody>
</table>
Configuring Threat Detection

This chapter describes how to configure threat detection statistics and scanning threat detection and includes the following sections:

- Information About Threat Detection, page 47-1
- Licensing Requirements for Threat Detection, page 47-1
- Configuring Basic Threat Detection Statistics, page 47-2
- Configuring Advanced Threat Detection Statistics, page 47-6
- Configuring Scanning Threat Detection, page 47-15
- Configuration Examples for Threat Detection, page 47-19

Information About Threat Detection

The threat detection feature consists of the following elements:

- Different levels of statistics gathering for various threats.

Threat detection statistics can help you manage threats to your ASASM; for example, if you enable scanning threat detection, then viewing statistics can help you analyze the threat. You can configure two types of threat detection statistics:

- Basic threat detection statistics—Includes information about attack activity for the system as a whole. Basic threat detection statistics are enabled by default and have no performance impact.
- Advanced threat detection statistics—Tracks activity at an object level, so the ASASM can report activity for individual hosts, ports, protocols, or access lists. Advanced threat detection statistics can have a major performance impact, depending on the statistics gathered, so only the access list statistics are enabled by default.

- Scanning threat detection, which determines when a host is performing a scan.

You can optionally shun any hosts determined to be a scanning threat.

Licensing Requirements for Threat Detection

The following table shows the licensing requirements for this feature:
Chapter 47      Configuring Threat Detection

Configuring Basic Threat Detection Statistics

Basic threat detection statistics include activity that might be related to an attack, such as a DoS attack. This section includes the following topics:

- Information About Basic Threat Detection Statistics, page 47-2
- Guidelines and Limitations, page 47-3
- Default Settings, page 47-3
- Configuring Basic Threat Detection Statistics, page 47-4
- Feature History for Basic Threat Detection Statistics, page 47-6

Information About Basic Threat Detection Statistics

Using basic threat detection statistics, the ASASM monitors the rate of dropped packets and security events due to the following reasons:

- Denial by access lists
- Bad packet format (such as invalid-ip-header or invalid-tcp-hdr-length)
- Connection limits exceeded (both system-wide resource limits, and limits set in the configuration)
- DoS attack detected (such as an invalid SPI, Stateful Firewall check failure)
- Basic firewall checks failed (This option is a combined rate that includes all firewall-related packet drops in this bulleted list. It does not include non-firewall-related drops such as interface overload, packets failed at application inspection, and scanning attack detected.)
- Suspicious ICMP packets detected
- Packets failed application inspection
- Interface overload
- Scanning attack detected (This option monitors scanning attacks; for example, the first TCP packet is not a SYN packet, or the TCP connection failed the 3-way handshake. Full scanning threat detection (see the “Configuring Scanning Threat Detection” section on page 47-15) takes this scanning attack rate information and acts on it by classifying hosts as attackers and automatically shunning them, for example.)
- Incomplete session detection such as TCP SYN attack detected or no data UDP session attack detected

When the ASASM detects a threat, it immediately sends a system log message (733100). The ASASM tracks two types of rates: the average event rate over an interval, and the burst event rate over a shorter burst interval. The burst rate interval is 1/30th of the average rate interval or 10 seconds, whichever is
higher. For each received event, the ASASM checks the average and burst rate limits; if both rates are exceeded, then the ASASM sends two separate system messages, with a maximum of one message for each rate type per burst period.

Basic threat detection affects performance only when there are drops or potential threats; even in this scenario, the performance impact is insignificant.

**Guidelines and Limitations**

This section includes the guidelines and limitations for this feature:

**Security Context Guidelines**
Supported in single mode only. Multiple mode is not supported.

**Firewall Mode Guidelines**
Supported in routed and transparent firewall mode.

**Types of Traffic Monitored**
Only through-the-box traffic is monitored; to-the-box traffic is not included in threat detection.

**Default Settings**

Basic threat detection statistics are enabled by default.

Table 47-1 lists the default settings. You can view all these default settings using the `show running-config all threat-detection` command.

<table>
<thead>
<tr>
<th>Packet Drop Reason</th>
<th>Trigger Settings</th>
<th>Burst Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Average Rate</td>
<td></td>
</tr>
<tr>
<td>DoS attack detected</td>
<td>100 drops/sec over the last 600 seconds.</td>
<td>400 drops/sec over the last 20 second period.</td>
</tr>
<tr>
<td>Bad packet format</td>
<td>80 drops/sec over the last 3600 seconds.</td>
<td>320 drops/sec over the last 120 second period.</td>
</tr>
<tr>
<td>Connection limits exceeded</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Suspicious ICMP packets detected</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Scanning attack detected</td>
<td>5 drops/sec over the last 600 seconds.</td>
<td>10 drops/sec over the last 20 second period.</td>
</tr>
<tr>
<td>Incomplete session detected such as TCP SYN attack detected or no data UDP session attack detected (combined)</td>
<td>100 drops/sec over the last 600 seconds.</td>
<td>200 drops/sec over the last 20 second period.</td>
</tr>
<tr>
<td></td>
<td>80 drops/sec over the last 3600 seconds.</td>
<td>160 drops/sec over the last 120 second period.</td>
</tr>
</tbody>
</table>
Table 47-1 Basic Threat Detection Default Settings (continued)

<table>
<thead>
<tr>
<th>Packet Drop Reason</th>
<th>Trigger Settings</th>
<th>Average Rate</th>
<th>Burst Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Denial by access lists</td>
<td></td>
<td>400 drops/sec over the last 600 seconds.</td>
<td>800 drops/sec over the last 20 second period.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>320 drops/sec over the last 3600 seconds.</td>
<td>640 drops/sec over the last 120 second period.</td>
</tr>
<tr>
<td>• Basic firewall checks failed</td>
<td></td>
<td>400 drops/sec over the last 600 seconds.</td>
<td>1600 drops/sec over the last 20 second period.</td>
</tr>
<tr>
<td>• Packets failed application inspection</td>
<td></td>
<td>320 drops/sec over the last 3600 seconds.</td>
<td>1280 drops/sec over the last 120 second period.</td>
</tr>
<tr>
<td>Interface overload</td>
<td></td>
<td>2000 drops/sec over the last 600 seconds.</td>
<td>8000 drops/sec over the last 20 second period.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1600 drops/sec over the last 3600 seconds.</td>
<td>6400 drops/sec over the last 120 second period.</td>
</tr>
</tbody>
</table>

Configuring Basic Threat Detection Statistics

This section describes how to configure basic threat detection statistics, including enabling or disabling it and changing the default limits.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 threat-detection basic-threat</td>
<td>Enables basic threat detection statistics (if you previously disabled it). Basic threat detection is enabled by default.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# threat-detection</td>
<td></td>
</tr>
<tr>
<td>basic-threat</td>
<td></td>
</tr>
<tr>
<td>Step 2 threat-detection rate {acl-drop</td>
<td>bad-packet-drop</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# threat-detection rate</td>
<td></td>
</tr>
<tr>
<td>dos-drop rate-interval 600 average-rate 60</td>
<td></td>
</tr>
<tr>
<td>burst-rate 100</td>
<td></td>
</tr>
</tbody>
</table>
Monitoring Basic Threat Detection Statistics

To monitor basic threat detection statistics, perform one of the following tasks:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show threat-detection rate</code>&lt;br&gt;&lt;br&gt;[<code>min-display-rate min_display_rate</code>]&lt;br&gt;`[acl-drop</td>
<td>bad-packet-drop</td>
</tr>
<tr>
<td><code>clear threat-detection rate</code></td>
<td>Clears basic threat statistics.</td>
</tr>
</tbody>
</table>

Examples

The following is sample output from the `show threat-detection rate` command:

```
hostname# show threat-detection rate

Average(eps) Current(eps) Trigger Total events
10-min ACL drop: 0 0 0 16
1-hour ACL drop: 0 0 0 112
1-hour SYN attck: 5 0 2 21438
10-min Scanning: 0 0 29 193
1-hour Scanning: 106 0 10 384776
1-hour Bad pkts: 76 0 2 274690
10-min Firewall: 0 0 3 22
1-hour Firewall: 76 0 2 274844
10-min DoS attck: 0 0 0 6
1-hour DoS attck: 0 0 0 42
10-min Interface: 0 0 0 204
1-hour Interface: 88 0 0 318225
```
Feature History for Basic Threat Detection Statistics

Table 47-2 lists each feature change and the platform release in which it was implemented.

Table 47-2  Feature History for Basic Threat Detection Statistics

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic threat detection statistics</td>
<td>8.0(2)</td>
<td>Basic threat detection statistics was introduced.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The following commands were introduced: threat-detection basic-threat, threat-detection rate, show threat-detection rate, clear threat-detection rate.</td>
</tr>
<tr>
<td>Burst rate interval changed to 1/30th of the average rate.</td>
<td>8.2(1)</td>
<td>In earlier releases, the burst rate interval was 1/60th of the average rate. To maximize memory usage, the sampling interval was reduced to 30 times during the average rate.</td>
</tr>
<tr>
<td>Improved memory usage</td>
<td>8.3(1)</td>
<td>The memory usage for threat detection was improved.</td>
</tr>
</tbody>
</table>

Configuring Advanced Threat Detection Statistics

You can configure the ASASM to collect extensive statistics. This section includes the following topics:

- Information About Advanced Threat Detection Statistics, page 47-6
- Guidelines and Limitations, page 47-6
- Default Settings, page 47-7
- Configuring Advanced Threat Detection Statistics, page 47-7
- Monitoring Advanced Threat Detection Statistics, page 47-9
- Feature History for Advanced Threat Detection Statistics, page 47-14

Information About Advanced Threat Detection Statistics

Advanced threat detection statistics show both allowed and dropped traffic rates for individual objects such as hosts, ports, protocols, or access lists.

Caution

Enabling advanced statistics can affect the ASASM performance, depending on the type of statistics enabled. The threat-detection statistics host command affects performance in a significant way; if you have a high traffic load, you might consider enabling this type of statistics temporarily. The threat-detection statistics port command, however, has modest impact.

Guidelines and Limitations

This section includes the guidelines and limitations for this feature:
Security Context Guidelines
Only TCP Intercept statistics are available in multiple mode.

Firewall Mode Guidelines
Supported in routed and transparent firewall mode.

Types of Traffic Monitored
Only through-the-box traffic is monitored; to-the-box traffic is not included in threat detection.

Default Settings
By default, statistics for access lists are enabled.

Configuring Advanced Threat Detection Statistics
By default, statistics for access lists are enabled. To enable other statistics, perform the following steps.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>threat-detection statistics</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# threat-detection statistics</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>threat-detection statistics access-list</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# threat-detection statistics access-list</td>
</tr>
</tbody>
</table>
### Command

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 3**

threat-detection statistics host [number-of-rate (1 | 2 | 3)] | (Optional) Enables statistics for hosts.  
The **number-of-rate** keyword sets the number of rate intervals maintained for host statistics. The default number of rate intervals is 1, which keeps the memory usage low. To view more rate intervals, set the value to 2 or 3. For example, if you set the value to 3, then you view data for the last 1 hour, 8 hours, and 24 hours. If you set this keyword to 1 (the default), then only the shortest rate interval statistics are maintained. If you set the value to 2, then the two shortest intervals are maintained.  
The host statistics accumulate for as long as the host is active and in the scanning threat host database. The host is deleted from the database (and the statistics cleared) after 10 minutes of inactivity. |

**Example:**

```
hostname(config)# threat-detection statistics host number-of-rate 2
```

| **Step 4**

threat-detection statistics port [number-of-rate (1 | 2 | 3)] | (Optional) Enables statistics for TCP and UDP ports.  
The **number-of-rate** keyword sets the number of rate intervals maintained for port statistics. The default number of rate intervals is 1, which keeps the memory usage low. To view more rate intervals, set the value to 2 or 3. For example, if you set the value to 3, then you view data for the last 1 hour, 8 hours, and 24 hours. If you set this keyword to 1 (the default), then only the shortest rate interval statistics are maintained. If you set the value to 2, then the two shortest intervals are maintained. |

**Example:**

```
hostname(config)# threat-detection statistics port number-of-rate 2
```
Chapter 47 Configuring Threat Detection

Configuring Advanced Threat Detection Statistics

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 5 threat-detection statistics protocol [number-of-rate {1 \mid 2 \mid 3}]</td>
<td>(Optional) Enables statistics for non-TCP/UDP IP protocols. The number-of-rate keyword sets the number of rate intervals maintained for protocol statistics. The default number of rate intervals is 1, which keeps the memory usage low. To view more rate intervals, set the value to 2 or 3. For example, if you set the value to 3, then you view data for the last 1 hour, 8 hours, and 24 hours. If you set this keyword to 1 (the default), then only the shortest rate interval statistics are maintained. If you set the value to 2, then the two shortest intervals are maintained.</td>
</tr>
<tr>
<td>Example: hostname(config)# threat-detection statistics protocol number-of-rate 3</td>
<td></td>
</tr>
<tr>
<td>Step 6 threat-detection statistics tcp-intercept [rate-interval minutes] [burst-rate attacks_per_sec] [average-rate attacks_per_sec]</td>
<td>(Optional) Enables statistics for attacks intercepted by TCP Intercept (see the Chapter 44, “Configuring Connection Settings,” to enable TCP Intercept). The rate-interval keyword sets the size of the history monitoring window, between 1 and 1440 minutes. The default is 30 minutes. During this interval, the ASASM samples the number of attacks 30 times. The burst-rate keyword sets the threshold for syslog message generation, between 25 and 2147483647. The default is 400 per second. When the burst rate is exceeded, syslog message 733104 is generated. The average-rate keyword sets the average rate threshold for syslog message generation, between 25 and 2147483647. The default is 200 per second. When the average rate is exceeded, syslog message 733105 is generated.</td>
</tr>
<tr>
<td>Example: hostname(config)# threat-detection statistics tcp-intercept rate-interval 60 burst-rate 800 average-rate 600</td>
<td></td>
</tr>
</tbody>
</table>

### Monitoring Advanced Threat Detection Statistics

The display output shows the following:

- The average rate in events/sec over fixed time periods.
- The current burst rate in events/sec over the last completed burst interval, which is 1/30th of the average rate interval or 10 seconds, whichever is larger.
- The number of times the rates were exceeded (for dropped traffic statistics only)
- The total number of events over the fixed time periods.
The ASASM stores the count at the end of each burst period, for a total of 30 completed burst intervals. The unfinished burst interval presently occurring is not included in the average rate. For example, if the average rate interval is 20 minutes, then the burst interval is 20 seconds. If the last burst interval was from 3:00:00 to 3:00:20, and you use the `show` command at 3:00:25, then the last 5 seconds are not included in the output.

The only exception to this rule is if the number of events in the unfinished burst interval already exceeds the number of events in the oldest burst interval (#1 of 30) when calculating the total events. In that case, the ASASM calculates the total events as the last 29 complete intervals, plus the events so far in the unfinished burst interval. This exception lets you monitor a large increase in events in real time.
To monitor advanced threat detection statistics, perform one of the following tasks:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show threat-detection statistics</code></td>
<td>Displays the top 10 statistics.</td>
</tr>
<tr>
<td><code>min-display-rate min_display_rate</code></td>
<td>The <code>min-display-rate min_display_rate</code> argument limits the display to statistics that exceed the minimum display rate in events per second. You can set the <code>min_display_rate</code> between 0 and 2147483647.</td>
</tr>
<tr>
<td><code>top</code></td>
<td>If you do not enter any options, the top 10 statistics are shown for all categories.</td>
</tr>
<tr>
<td>`access-list</td>
<td>host</td>
</tr>
<tr>
<td>`rate-1</td>
<td>rate-2</td>
</tr>
<tr>
<td><code>tcp-intercept [all] detail</code></td>
<td>To view statistics for ports and protocols, use the <code>port-protocol</code> keyword. The <code>port-protocol</code> keyword shows statistics for both ports and protocols (both must be enabled for the display), and shows the combined statistics of TCP/UDP port and IP protocol types. TCP (protocol 6) and UDP (protocol 17) are not included in the display for IP protocols; TCP and UDP ports are, however, included in the display for ports. If you only enable statistics for one of these types, port or protocol, then you will only view the enabled statistics.</td>
</tr>
<tr>
<td><code>show threat-detection statistics</code></td>
<td>To view TCP Intercept statistics, use the <code>tcp-intercept</code> keyword. The display includes the top 10 protected servers under attack. The <code>all</code> keyword shows the history data of all the traced servers. The <code>detail</code> keyword shows history sampling data. The ASASM samples the number of attacks 30 times during the rate interval, so for the default 30 minute period, statistics are collected every 60 seconds.</td>
</tr>
<tr>
<td><code>min-display-rate min_display_rate</code></td>
<td>The rate-1 keyword shows the statistics for the smallest fixed rate intervals available in the display; rate-2 shows the next largest rate interval; and rate-3, if you have three intervals defined, shows the largest rate interval. For example, the display shows statistics for the last 1 hour, 8 hours, and 24 hours. If you set the rate-1 keyword, the ASASM shows only the 1 hour time interval.</td>
</tr>
</tbody>
</table>

show threat-detection statistics
[min-display-rate min_display_rate] host
[ip_address [mask]]

Displays statistics for all hosts or for a specific host or subnet.

show threat-detection statistics
[min-display-rate min_display_rate] port
[start_port[-end_port]]

Displays statistics for all ports or for a specific port or range of ports.
Chapter 47  Configuring Threat Detection

Configuring Advanced Threat Detection Statistics

Examples

The following is sample output from the `show threat-detection statistics host` command:

```
hostname# show threat-detection statistics host

Average(eps)  Current(eps)  Trigger  Total events
Host:10.0.0.1: tot-ses:289235 act-ses:22571 fw-drop:0 insp-drop:0 null-ses:21438 bad-acc:0
  1-hour Sent byte: 2938 0 0 10580308
  8-hour Sent byte: 367 0 0 10580308
  24-hour Sent byte: 122 0 0 10580308
  1-hour Sent pkts: 28 0 0 104043
  8-hour Sent pkts: 3 0 0 104043
  24-hour Sent pkts: 1 0 0 104043
  20-min Sent drop: 9 0 1 10851
  1-hour Sent drop: 3 0 1 10851
  1-hour Recv byte: 2697 0 0 9712670
  8-hour Recv byte: 337 0 0 9712670
  24-hour Recv byte: 112 0 0 9712670
  1-hour Recv pkts: 29 0 0 104846
  8-hour Recv pkts: 3 0 0 104846
  24-hour Recv pkts: 1 0 0 104846
  20-min Recv drop: 42 0 3 50567
  1-hour Recv drop: 14 0 1 50567

Host:10.0.0.0: tot-ses:1 act-ses:0 fw-drop:0 insp-drop:0 null-ses:0 bad-acc:0
  1-hour Sent byte: 0 0 0 614
  8-hour Sent byte: 0 0 0 614
  24-hour Sent byte: 0 0 0 614
  1-hour Sent pkts: 0 0 0 6
  8-hour Sent pkts: 0 0 0 6
  24-hour Sent pkts: 0 0 0 6
  20-min Sent drop: 0 0 0 4
  1-hour Sent drop: 0 0 0 4
  1-hour Recv byte: 0 0 0 706
  8-hour Recv byte: 0 0 0 706
  24-hour Recv byte: 0 0 0 706
  1-hour Recv pkts: 0 0 0 7
```

Table 47-3 shows each field description.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host</td>
<td>Shows the host IP address.</td>
</tr>
<tr>
<td>tot-ses</td>
<td>Shows the total number of sessions for this host since it was added to the database.</td>
</tr>
<tr>
<td>act-ses</td>
<td>Shows the total number of active sessions that the host is currently involved in.</td>
</tr>
</tbody>
</table>
**Table 47-3  show threat-detection statistics host Command Fields (continued)**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>fw-drop</td>
<td>Shows the number of firewall drops. Firewall drops is a combined rate that includes all firewall-related packet drops tracked in basic threat detection, including access list denials, bad packets, exceeded connection limits, DoS attack packets, suspicious ICMP packets, TCP SYN attack packets, and no data UDP attack packets. It does not include non-firewall-related drops such as interface overload, packets failed at application inspection, and scanning attack detected.</td>
</tr>
<tr>
<td>insp-drop</td>
<td>Shows the number of packets dropped because they failed application inspection.</td>
</tr>
<tr>
<td>null-ses</td>
<td>Shows the number of null sessions, which are TCP SYN sessions that did not complete within the 3-second timeout, and UDP sessions that did not have any data sent by its server 3 seconds after the session starts.</td>
</tr>
<tr>
<td>bad-acc</td>
<td>Shows the number of bad access attempts to host ports that are in a closed state. When a port is determined to be in a null session (see the null-ses field description), the port state of the host is set to HOST_PORT_CLOSE. Any client accessing the port of the host is immediately classified as a bad access without the need to wait for a timeout.</td>
</tr>
<tr>
<td>Average(eps)</td>
<td>Shows the average rate in events/sec over each time period. The ASASM stores the count at the end of each burst period, for a total of 30 completed burst intervals. The unfinished burst interval presently occurring is not included in the average rate. For example, if the average rate interval is 20 minutes, then the burst interval is 20 seconds. If the last burst interval was from 3:00:00 to 3:00:20, and you use the show command at 3:00:25, then the last 5 seconds are not included in the output. The only exception to this rule is if the number of events in the unfinished burst interval already exceeds the number of events in the oldest burst interval (#1 of 30) when calculating the total events. In that case, the ASASM calculates the total events as the last 29 complete intervals, plus the events so far in the unfinished burst interval. This exception lets you monitor a large increase in events in real time.</td>
</tr>
<tr>
<td>Current(eps)</td>
<td>Shows the current burst rate in events/sec over the last completed burst interval, which is 1/30th of the average rate interval or 10 seconds, whichever is larger. For the example specified in the Average(eps) description, the current rate is the rate from 3:19:30 to 3:20:00</td>
</tr>
<tr>
<td>Trigger</td>
<td>Shows the number of times the dropped packet rate limits were exceeded. For valid traffic identified in the sent and received bytes and packets rows, this value is always 0, because there are no rate limits to trigger for valid traffic.</td>
</tr>
<tr>
<td>Total events</td>
<td>Shows the total number of events over each rate interval. The unfinished burst interval presently occurring is not included in the total events. The only exception to this rule is if the number of events in the unfinished burst interval already exceeds the number of events in the oldest burst interval (#1 of 30) when calculating the total events. In that case, the ASASM calculates the total events as the last 29 complete intervals, plus the events so far in the unfinished burst interval. This exception lets you monitor a large increase in events in real time.</td>
</tr>
</tbody>
</table>
### Table 47-3  show threat-detection statistics host Command Fields (continued)

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>20-min, 1-hour, 8-hour, and 24-hour</td>
<td>Shows statistics for these fixed rate intervals.</td>
</tr>
<tr>
<td>Sent byte</td>
<td>Shows the number of successful bytes sent from the host.</td>
</tr>
<tr>
<td>Sent pkts</td>
<td>Shows the number of successful packets sent from the host.</td>
</tr>
<tr>
<td>Sent drop</td>
<td>Shows the number of packets sent from the host that were dropped because they were part of a scanning attack.</td>
</tr>
<tr>
<td>Recv byte</td>
<td>Shows the number of successful bytes received by the host.</td>
</tr>
<tr>
<td>Recv pkts</td>
<td>Shows the number of successful packets received by the host.</td>
</tr>
<tr>
<td>Recv drop</td>
<td>Shows the number of packets received by the host that were dropped because they were part of a scanning attack.</td>
</tr>
</tbody>
</table>

### Feature History for Advanced Threat Detection Statistics

Table 47-4 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Advanced threat detection statistics</td>
<td>8.0(2)</td>
<td>Advanced threat detection statistics was introduced. The following commands were introduced: <code>threat-detection statistics</code>, <code>show threat-detection statistics</code>.</td>
</tr>
<tr>
<td>TCP Intercept statistics</td>
<td>8.0(4)/8.1(2)</td>
<td>TCP Intercept statistics were introduced. The following commands were modified or introduced: <code>threat-detection statistics tcp-intercept</code>, <code>show threat-detection statistics top tcp-intercept</code>, <code>clear threat-detection statistics</code>.</td>
</tr>
<tr>
<td>Customize host statistics rate intervals</td>
<td>8.1(2)</td>
<td>You can now customize the number of rate intervals for which statistics are collected. The default number of rates was changed from 3 to 1. The following command was modified: <code>threat-detection statistics host number-of-rates</code>.</td>
</tr>
<tr>
<td>Burst rate interval changed to 1/30th of the average rate.</td>
<td>8.2(1)</td>
<td>In earlier releases, the burst rate interval was 1/60th of the average rate. To maximize memory usage, the sampling interval was reduced to 30 times during the average rate.</td>
</tr>
</tbody>
</table>
Chapter 47      Configuring Threat Detection

Configuring Scanning Threat Detection

This section includes the following topics:

- Information About Scanning Threat Detection, page 47-15
- Guidelines and Limitations, page 47-16
- Default Settings, page 47-16
- Configuring Scanning Threat Detection, page 47-17
- Monitoring Shunned Hosts, Attackers, and Targets, page 47-17

Information About Scanning Threat Detection

A typical scanning attack consists of a host that tests the accessibility of every IP address in a subnet (by scanning through many hosts in the subnet or sweeping through many ports in a host or subnet). The scanning threat detection feature determines when a host is performing a scan. Unlike IPS scan detection that is based on traffic signatures, the ASASM scanning threat detection feature maintains an extensive database that contains host statistics that can be analyzed for scanning activity.

The host database tracks suspicious activity such as connections with no return activity, access of closed service ports, vulnerable TCP behaviors such as non-random IPID, and many more behaviors.

If the scanning threat rate is exceeded, then the ASASM sends a syslog message (733101), and optionally shuns the attacker. The ASASM tracks two types of rates: the average event rate over an interval, and the burst event rate over a shorter burst interval. The burst event rate is 1/30th of the average rate interval or 10 seconds, whichever is higher. For each event detected that is considered to be part of a scanning attack, the ASASM checks the average and burst rate limits. If either rate is exceeded for traffic sent from a host, then that host is considered to be an attacker. If either rate is exceeded for traffic received by a host, then that host is considered to be a target.

Caution

The scanning threat detection feature can affect the ASASM performance and memory significantly while it creates and gathers host- and subnet-based data structure and information.

Table 47-4  Feature History for Advanced Threat Detection Statistics (continued)

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customize port and protocol statistics rate intervals</td>
<td>8.3(1)</td>
<td>You can now customize the number of rate intervals for which statistics are collected. The default number of rates was changed from 3 to 1.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The following commands were modified: <code>threat-detection statistics port number-of-rates</code>, <code>threat-detection statistics protocol number-of-rates</code>.</td>
</tr>
<tr>
<td>Improved memory usage</td>
<td>8.3(1)</td>
<td>The memory usage for threat detection was improved. The following command was introduced: <code>show threat-detection memory</code>.</td>
</tr>
</tbody>
</table>
Guidelines and Limitations

This section includes the guidelines and limitations for this feature:

Security Context Guidelines
Supported in single mode only. Multiple mode is not supported.

Firewall Mode Guidelines
Supported in routed and transparent firewall mode.

Types of Traffic Monitored
- Only through-the-box traffic is monitored; to-the-box traffic is not included in threat detection.
- Traffic that is denied by an access list does not trigger scanning threat detection; only traffic that is allowed through the ASASM and that creates a flow is affected by scanning threat detection.

Default Settings

Table 47-5 lists the default rate limits for scanning threat detection.

<table>
<thead>
<tr>
<th>Average Rate</th>
<th>Burst Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 drops/sec over the last 600 seconds.</td>
<td>10 drops/sec over the last 20 second period.</td>
</tr>
<tr>
<td>5 drops/sec over the last 3600 seconds.</td>
<td>10 drops/sec over the last 120 second period.</td>
</tr>
</tbody>
</table>

The burst rate is calculated as the average rate every $N$ seconds, where $N$ is the burst rate interval. The burst rate interval is $1/30$th of the rate interval or 10 seconds, whichever is larger.
Configuring Scanning Threat Detection

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Enables scanning threat detection. By default, the system log message 733101 is generated when a host is identified as an attacker. Enter this command multiple times to identify multiple IP addresses or network object groups to exempt from shunning.</td>
</tr>
<tr>
<td>`threat-detection scanning-threat</td>
<td>shun</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>`hostname(config)# threat-detection</td>
<td>scanning-threat</td>
</tr>
</tbody>
</table>

| **Step 2** | (Optional) Sets the duration of the shun for attacking hosts. |
| `threat-detection scanning-threat shun |duration |seconds` | |
| **Example:** | |
| `hostname(config)# threat-detection |scanning-threat |shun |duration |2000` | |

| **Step 3** | (Optional) Changes the default event limit for when the ASASM identifies a host as an attacker or as a target. If you already configured this command as part of the basic threat detection configuration (see the “Configuring Basic Threat Detection Statistics” section on page 47-2), then those settings are shared with the scanning threat detection feature; you cannot configure separate rates for basic and scanning threat detection. If you do not set the rates using this command, the default values are used for both the scanning threat detection feature and the basic threat detection feature. You can configure up to three different rate intervals, by entering separate commands. |
| `threat-detection rate scanning-threat |rate-interval |rate_interval |average-rate |av_rate |burst-rate |burst_rate` | |
| **Example:** | |
| `hostname(config)# threat-detection |rate |scanning-threat |rate-interval |1200 |average-rate |10 |burst-rate |20` | |
| `hostname(config)# threat-detection |rate |scanning-threat |rate-interval |2400 |average-rate |10 |burst-rate |20` | |

Monitoring Shunned Hosts, Attackers, and Targets

To monitor shunned hosts and attackers and targets, perform one of the following tasks:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show threat-detection shun</code></td>
<td>Displays the hosts that are currently shunned.</td>
</tr>
</tbody>
</table>
Configuring Scanning Threat Detection

### Command Syntax

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>clear threat-detection shun [ip_address [mask]]</code></td>
<td>Releases a host from being shunned. If you do not specify an IP address, all hosts are cleared from the shun list.</td>
</tr>
<tr>
<td>`show threat-detection scanning-threat [attacker</td>
<td>target]`</td>
</tr>
</tbody>
</table>

### Examples

The following is sample output from the `show threat-detection shun` command:

```
hostname# show threat-detection shun
Shunned Host List:
10.1.1.6
192.168.6.7
```

To release the host at 10.1.1.6, enter the following command:

```
hostname# clear threat-detection shun 10.1.1.6
```

The following is sample output from the `show threat-detection scanning-threat attacker` command:

```
hostname# show threat-detection scanning-threat attacker
10.1.2.3
10.8.3.6
209.165.200.225
```

### Feature History for Scanning Threat Detection

Table 47-6  lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scanning threat detection</td>
<td>8.0(2)</td>
<td>Scanning threat detection was introduced. The following commands were introduced: <code>threat-detection scanning-threat</code>, <code>threat-detection rate scanning-threat</code>, <code>show threat-detection scanning-threat</code>, <code>show threat-detection shun</code>, <code>clear threat-detection shun</code>.</td>
</tr>
<tr>
<td>Shun duration</td>
<td>8.0(4)/8.1(2)</td>
<td>You can now set the shun duration, The following command was introduced: <code>threat-detection scanning-threat shun duration</code>.</td>
</tr>
</tbody>
</table>
Configuration Examples for Threat Detection

The following example configures basic threat detection statistics, and changes the DoS attack rate settings. All advanced threat detection statistics are enabled, with the host statistics number of rate intervals lowered to 2. The TCP Intercept rate interval is also customized. Scanning threat detection is enabled with automatic shunning for all addresses except 10.1.1.0/24. The scanning threat rate intervals are customized.

```
threat-detection basic-threat
threat-detection rate dos-drop rate-interval 600 average-rate 60 burst-rate 100
threat-detection statistics
threat-detection statistics host number-of-rate 2
threat-detection statistics tcp-intercept rate-interval 60 burst-rate 800 average-rate 600
threat-detection scanning-threat shun except ip-address 10.1.1.0 255.255.255.0
threat-detection rate scanning-threat rate-interval 1200 average-rate 10 burst-rate 20
threat-detection rate scanning-threat rate-interval 2400 average-rate 10 burst-rate 20
```

---

Table 47-6 Feature History for Scanning Threat Detection (continued)

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Burst rate interval changed to 1/30th of the average rate.</td>
<td>8.2(1)</td>
<td>In earlier releases, the burst rate interval was 1/60th of the average rate. To maximize memory usage, the sampling interval was reduced to 30 times during the average rate.</td>
</tr>
<tr>
<td>Improved memory usage</td>
<td>8.3(1)</td>
<td>The memory usage for threat detection was improved.</td>
</tr>
</tbody>
</table>
Using Protection Tools

This chapter describes some of the many tools available to protect your network and includes the following sections:

- Preventing IP Spoofing, page 48-1
- Configuring the Fragment Size, page 48-2
- Blocking Unwanted Connections, page 48-2
- Configuring IP Audit for Basic IPS Support, page 48-3

Preventing IP Spoofing

This section lets you enable Unicast Reverse Path Forwarding on an interface. Unicast RPF guards against IP spoofing (a packet uses an incorrect source IP address to obscure its true source) by ensuring that all packets have a source IP address that matches the correct source interface according to the routing table.

Normally, the ASASM only looks at the destination address when determining where to forward the packet. Unicast RPF instructs the ASASM to also look at the source address; this is why it is called Reverse Path Forwarding. For any traffic that you want to allow through the ASASM, the ASASM routing table must include a route back to the source address. See RFC 2267 for more information.

For outside traffic, for example, the ASASM can use the default route to satisfy the Unicast RPF protection. If traffic enters from an outside interface, and the source address is not known to the routing table, the ASASM uses the default route to correctly identify the outside interface as the source interface.

If traffic enters the outside interface from an address that is known to the routing table, but is associated with the inside interface, then the ASASM drops the packet. Similarly, if traffic enters the inside interface from an unknown source address, the ASASM drops the packet because the matching route (the default route) indicates the outside interface.

Unicast RPF is implemented as follows:

- ICMP packets have no session, so each packet is checked.
- UDP and TCP have sessions, so the initial packet requires a reverse route lookup. Subsequent packets arriving during the session are checked using an existing state maintained as part of the session. Non-initial packets are checked to ensure they arrived on the same interface used by the initial packet.

To enable Unicast RPF, enter the following command:

```
hostname(config)# ip verify reverse-path interface interface_name
```
Configuring the Fragment Size

By default, the ASASM allows up to 24 fragments per IP packet, and up to 200 fragments awaiting reassembly. You might need to let fragments on your network if you have an application that routinely fragments packets, such as NFS over UDP. However, if you do not have an application that fragments traffic, we recommend that you do not allow fragments through the ASASM. Fragmented packets are often used as DoS attacks.

To set disallow fragments, enter the following command:

```
hostname(config)# fragment chain 1 [interface_name]
```

Enter an interface name if you want to prevent fragmentation on a specific interface. By default, this command applies to all interfaces.

Blocking Unwanted Connections

If you know that a host is attempting to attack your network (for example, syslog messages show an attack), then you can block (or shun) connections based on the source IP address. All existing connections and new connections are blocked until you remove the shun.

**Note**

If you have an IPS that monitors traffic, such as an AIP SSM, then the IPS can shun connections automatically.

To shun a connection manually, perform the following steps:

**Step 1**
If necessary, view information about the connection by entering the following command:

```
hostname# show conn
```

The ASASM shows information about each connection, such as the following:

TCP out 64.101.68.161:4300 in 10.86.194.60:23 idle 0:00:00 bytes 1297 flags UIO

**Step 2**
To shun connections from the source IP address, enter the following command:

```
hostname(config)# shun src_ip [dst_ip src_port dest_port [protocol]] [vlan vlan_id]
```

If you enter only the source IP address, then all future connections are shunned; existing connections remain active.

To drop an existing connection, as well as blocking future connections from the source IP address, enter the destination IP address, source and destination ports, and the protocol. By default, the protocol is 0 for IP. Note that specifying the additional parameters is a convenient way to also drop a specific current connection; the shun, however, remains in place for all future connections from the source IP address, regardless of destination parameters.

For multiple context mode, you can enter this command in the admin context, and by specifying a VLAN ID that is assigned to an interface in other contexts, you can shun the connection in other contexts.

**Step 3**
To remove the shun, enter the following command:

```
hostname(config)# no shun src_ip [vlan vlan_id]
```
Configuring IP Audit for Basic IPS Support

The IP audit feature provides basic IPS support for the ASASM that does not have an AIPS SSM. It supports a basic list of signatures, and you can configure the ASASM to perform one or more actions on traffic that matches a signature.

This section includes the following topics:
- Configuring IP Audit, page 48-3
- IP Audit Signature List, page 48-4

Configuring IP Audit

To enable IP audit, perform the following steps:

**Step 1** To define an IP audit policy for informational signatures, enter the following command:

```bash
hostname(config)# ip audit name name info [action [alarm] [drop] [reset]]
```

Where **alarm** generates a system message showing that a packet matched a signature, **drop** drops the packet, and **reset** drops the packet and closes the connection. If you do not define an action, then the default action is to generate an alarm.

**Step 2** To define an IP audit policy for attack signatures, enter the following command:

```bash
hostname(config)# ip audit name name attack [action [alarm] [drop] [reset]]
```

Where **alarm** generates a system message showing that a packet matched a signature, **drop** drops the packet, and **reset** drops the packet and closes the connection. If you do not define an action, then the default action is to generate an alarm.

**Step 3** To assign the policy to an interface, enter the following command:

```bash
ip audit interface interface_name policy_name
```

**Step 4** To disable signatures, or for more information about signatures, see the **ip audit signature** command in the command reference.
## IP Audit Signature List

Table 48-1 lists supported signatures and system message numbers.

**Table 48-1 Signature IDs and System Message Numbers**

<table>
<thead>
<tr>
<th>Signature ID</th>
<th>Message Number</th>
<th>Signature Title</th>
<th>Signature Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>400000</td>
<td>IP options-Bad Option List</td>
<td>Informational</td>
<td>Triggers on receipt of an IP datagram where the list of IP options in the IP datagram header is incomplete or malformed. The IP options list contains one or more options that perform various network management or debugging tasks.</td>
</tr>
<tr>
<td>1001</td>
<td>400001</td>
<td>IP options-Record Packet Route</td>
<td>Informational</td>
<td>Triggers on receipt of an IP datagram where the IP option list for the datagram includes option 7 (Record Packet Route).</td>
</tr>
<tr>
<td>1002</td>
<td>400002</td>
<td>IP options-Timestamp</td>
<td>Informational</td>
<td>Triggers on receipt of an IP datagram where the IP option list for the datagram includes option 4 (Timestamp).</td>
</tr>
<tr>
<td>1003</td>
<td>400003</td>
<td>IP options-Security</td>
<td>Informational</td>
<td>Triggers on receipt of an IP datagram where the IP option list for the datagram includes option 2 (Security options).</td>
</tr>
<tr>
<td>1004</td>
<td>400004</td>
<td>IP options-Loose Source Route</td>
<td>Informational</td>
<td>Triggers on receipt of an IP datagram where the IP option list for the datagram includes option 3 (Loose Source Route).</td>
</tr>
<tr>
<td>1005</td>
<td>400005</td>
<td>IP options-SATNET ID</td>
<td>Informational</td>
<td>Triggers on receipt of an IP datagram where the IP option list for the datagram includes option 8 (SATNET stream identifier).</td>
</tr>
<tr>
<td>1006</td>
<td>400006</td>
<td>IP options-Strict Source Route</td>
<td>Informational</td>
<td>Triggers on receipt of an IP datagram in which the IP option list for the datagram includes option 2 (Strict Source Routing).</td>
</tr>
<tr>
<td>1100</td>
<td>400007</td>
<td>IP Fragment Attack</td>
<td>Attack</td>
<td>Triggers when any IP datagram is received with an offset value less than 5 but greater than 0 indicated in the offset field.</td>
</tr>
<tr>
<td>1102</td>
<td>400008</td>
<td>IP Impossible Packet</td>
<td>Attack</td>
<td>Triggers when an IP packet arrives with source equal to destination address. This signature will catch the so-called Land Attack.</td>
</tr>
</tbody>
</table>
### Table 48-1 Signature IDs and System Message Numbers (continued)

<table>
<thead>
<tr>
<th>Signature ID</th>
<th>Message Number</th>
<th>Signature Title</th>
<th>Signature Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1103</td>
<td>400009</td>
<td>IP Overlapping Fragments (Teardrop)</td>
<td>Attack</td>
<td>Triggers when two fragments contained within the same IP datagram have offsets that indicate that they share positioning within the datagram. This could mean that fragment A is being completely overwritten by fragment B, or that fragment A is partially being overwritten by fragment B. Some operating systems do not properly handle fragments that overlap in this manner and may throw exceptions or behave in other undesirable ways upon receipt of overlapping fragments, which is how the Teardrop attack works to create a DoS.</td>
</tr>
<tr>
<td>2000</td>
<td>400010</td>
<td>ICMP Echo Reply</td>
<td>Informational</td>
<td>Triggers when a IP datagram is received with the protocol field of the IP header set to 1 (ICMP) and the type field in the ICMP header set to 0 (Echo Reply).</td>
</tr>
<tr>
<td>2001</td>
<td>400011</td>
<td>ICMP Host Unreachable</td>
<td>Informational</td>
<td>Triggers when an IP datagram is received with the protocol field of the IP header set to 1 (ICMP) and the type field in the ICMP header set to 3 (Host Unreachable).</td>
</tr>
<tr>
<td>2002</td>
<td>400012</td>
<td>ICMP Source Quench</td>
<td>Informational</td>
<td>Triggers when an IP datagram is received with the protocol field of the IP header set to 1 (ICMP) and the type field in the ICMP header set to 4 (Source Quench).</td>
</tr>
<tr>
<td>2003</td>
<td>400013</td>
<td>ICMP Redirect</td>
<td>Informational</td>
<td>Triggers when a IP datagram is received with the protocol field of the IP header set to 1 (ICMP) and the type field in the ICMP header set to 5 (Redirect).</td>
</tr>
<tr>
<td>2004</td>
<td>400014</td>
<td>ICMP Echo Request</td>
<td>Informational</td>
<td>Triggers when a IP datagram is received with the protocol field of the IP header set to 1 (ICMP) and the type field in the ICMP header set to 8 (Echo Request).</td>
</tr>
<tr>
<td>2005</td>
<td>400015</td>
<td>ICMP Time Exceeded for a Datagram</td>
<td>Informational</td>
<td>Triggers when a IP datagram is received with the protocol field of the IP header set to 1 (ICMP) and the type field in the ICMP header set to 11 (Time Exceeded for a Datagram).</td>
</tr>
<tr>
<td>2006</td>
<td>400016</td>
<td>ICMP Parameter Problem on Datagram</td>
<td>Informational</td>
<td>Triggers when a IP datagram is received with the protocol field of the IP header set to 1 (ICMP) and the type field in the ICMP header set to 12 (Parameter Problem on Datagram).</td>
</tr>
<tr>
<td>2007</td>
<td>400017</td>
<td>ICMP Timestamp Request</td>
<td>Informational</td>
<td>Triggers when a IP datagram is received with the protocol field of the IP header set to 1 (ICMP) and the type field in the ICMP header set to 13 (Timestamp Request).</td>
</tr>
</tbody>
</table>
### Table 48-1 Signature IDs and System Message Numbers (continued)

<table>
<thead>
<tr>
<th>Signature ID</th>
<th>Message Number</th>
<th>Signature Title</th>
<th>Signature Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2008</td>
<td>400018</td>
<td>ICMP Timestamp Reply</td>
<td>Informational</td>
<td>Triggers when a IP datagram is received with the protocol field of the IP header set to 1 (ICMP) and the type field in the ICMP header set to 14 (Timestamp Reply).</td>
</tr>
<tr>
<td>2009</td>
<td>400019</td>
<td>ICMP Information Request</td>
<td>Informational</td>
<td>Triggers when a IP datagram is received with the protocol field of the IP header set to 1 (ICMP) and the type field in the ICMP header set to 15 (Information Request).</td>
</tr>
<tr>
<td>2010</td>
<td>400020</td>
<td>ICMP Information Reply</td>
<td>Informational</td>
<td>Triggers when a IP datagram is received with the protocol field of the IP header set to 1 (ICMP) and the type field in the ICMP header set to 16 (Information Reply).</td>
</tr>
<tr>
<td>2011</td>
<td>400021</td>
<td>ICMP Address Mask Request</td>
<td>Informational</td>
<td>Triggers when a IP datagram is received with the protocol field of the IP header set to 1 (ICMP) and the type field in the ICMP header set to 17 (Address Mask Request).</td>
</tr>
<tr>
<td>2012</td>
<td>400022</td>
<td>ICMP Address Mask Reply</td>
<td>Informational</td>
<td>Triggers when a IP datagram is received with the protocol field of the IP header set to 1 (ICMP) and the type field in the ICMP header set to 18 (Address Mask Reply).</td>
</tr>
<tr>
<td>2150</td>
<td>400023</td>
<td>Fragmented ICMP Traffic</td>
<td>Attack</td>
<td>Triggers when a IP datagram is received with the protocol field of the IP header set to 1 (ICMP) and either the more fragments flag is set to 1 (ICMP) or there is an offset indicated in the offset field.</td>
</tr>
<tr>
<td>2151</td>
<td>400024</td>
<td>Large ICMP Traffic</td>
<td>Attack</td>
<td>Triggers when a IP datagram is received with the protocol field of the IP header set to 1 (ICMP) and the IP length &gt; 1024.</td>
</tr>
<tr>
<td>2154</td>
<td>400025</td>
<td>Ping of Death Attack</td>
<td>Attack</td>
<td>Triggers when a IP datagram is received with the protocol field of the IP header set to 1 (ICMP), the Last Fragment bit is set, and (IP offset * 8) + (IP data length) &gt; 65535 that is to say, the IP offset (which represents the starting position of this fragment in the original packet, and which is in 8 byte units) plus the rest of the packet is greater than the maximum size for an IP packet.</td>
</tr>
<tr>
<td>3040</td>
<td>400026</td>
<td>TCP NULL flags</td>
<td>Attack</td>
<td>Triggers when a single TCP packet with none of the SYN, FIN, ACK, or RST flags set has been sent to a specific host.</td>
</tr>
<tr>
<td>3041</td>
<td>400027</td>
<td>TCP SYN+FIN flags</td>
<td>Attack</td>
<td>Triggers when a single TCP packet with the SYN and FIN flags are set and is sent to a specific host.</td>
</tr>
<tr>
<td>1002</td>
<td>400002</td>
<td>IP options-Timestamp</td>
<td>Informational</td>
<td>Triggers on receipt of an IP datagram where the IP option list for the datagram includes option 4 (Timestamp).</td>
</tr>
<tr>
<td>------</td>
<td>--------</td>
<td>----------------------</td>
<td>---------------</td>
<td>---------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>1003</td>
<td>400003</td>
<td>IP options-Security</td>
<td>Informational</td>
<td>Triggers on receipt of an IP datagram where the IP option list for the datagram includes option 2 (Security options).</td>
</tr>
<tr>
<td>1004</td>
<td>400004</td>
<td>IP options-Loose Source Route</td>
<td>Informational</td>
<td>Triggers on receipt of an IP datagram where the IP option list for the datagram includes option 3 (Loose Source Route).</td>
</tr>
<tr>
<td>1005</td>
<td>400005</td>
<td>IP options-SATNET ID</td>
<td>Informational</td>
<td>Triggers on receipt of an IP datagram where the IP option list for the datagram includes option 8 (SATNET stream identifier).</td>
</tr>
<tr>
<td>1006</td>
<td>400006</td>
<td>IP options-Strict Source Route</td>
<td>Informational</td>
<td>Triggers on receipt of an IP datagram in which the IP option list for the datagram includes option 2 (Strict Source Routing).</td>
</tr>
<tr>
<td>1100</td>
<td>400007</td>
<td>IP Fragment Attack</td>
<td>Attack</td>
<td>Triggers when any IP datagram is received with an offset value less than 5 but greater than 0 indicated in the offset field.</td>
</tr>
<tr>
<td>1102</td>
<td>400008</td>
<td>IP Impossible Packet</td>
<td>Attack</td>
<td>Triggers when an IP packet arrives with source equal to destination address. This signature will catch the so-called Land Attack.</td>
</tr>
<tr>
<td>Signature ID</td>
<td>Message Number</td>
<td>Signature Title</td>
<td>Signature Type</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>----------------</td>
<td>-------------------------------------</td>
<td>----------------</td>
<td>-------------</td>
</tr>
<tr>
<td>2002</td>
<td>400012</td>
<td>ICMP Source Quench</td>
<td>Informational</td>
<td>Triggers when an IP datagram is received with the protocol field of the IP header set to 1 (ICMP) and the type field in the ICMP header set to 4 (Source Quench).</td>
</tr>
<tr>
<td>2003</td>
<td>400013</td>
<td>ICMP Redirect</td>
<td>Informational</td>
<td>Triggers when a IP datagram is received with the protocol field of the IP header set to 1 (ICMP) and the type field in the ICMP header set to 5 (Redirect).</td>
</tr>
<tr>
<td>2004</td>
<td>400014</td>
<td>ICMP Echo Request</td>
<td>Informational</td>
<td>Triggers when a IP datagram is received with the protocol field of the IP header set to 1 (ICMP) and the type field in the ICMP header set to 8 (Echo Request).</td>
</tr>
<tr>
<td>2005</td>
<td>400015</td>
<td>ICMP Time Exceeded for a Datagram</td>
<td>Informational</td>
<td>Triggers when a IP datagram is received with the protocol field of the IP header set to 1 (ICMP) and the type field in the ICMP header set to 11 (Time Exceeded for a Datagram).</td>
</tr>
<tr>
<td>2006</td>
<td>400016</td>
<td>ICMP Parameter Problem on Datagram</td>
<td>Informational</td>
<td>Triggers when a IP datagram is received with the protocol field of the IP header set to 1 (ICMP) and the type field in the ICMP header set to 12 (Parameter Problem on Datagram).</td>
</tr>
<tr>
<td>2007</td>
<td>400017</td>
<td>ICMP Timestamp Request</td>
<td>Informational</td>
<td>Triggers when a IP datagram is received with the protocol field of the IP header set to 1 (ICMP) and the type field in the ICMP header set to 13 (Timestamp Request).</td>
</tr>
</tbody>
</table>
Configuring IP Audit for Basic IPS Support

2150 400023  Fragmented ICMP Traffic  Attack  Triggers when a IP datagram is received with the protocol field of the IP header set to 1 (ICMP) and either the more fragments flag is set to 1 (ICMP) or there is an offset indicated in the offset field.

2151 400024  Large ICMP Traffic  Attack  Triggers when a IP datagram is received with the protocol field of the IP header set to 1 (ICMP) and the IP length > 1024.

2154 400025  Ping of Death Attack  Attack  Triggers when a IP datagram is received with the protocol field of the IP header set to 1 (ICMP), the Last Fragment bit is set, and (IP offset * 8) + (IP data length) > 65535 that is to say, the IP offset (which represents the starting position of this fragment in the original packet, and which is in 8 byte units) plus the rest of the packet is greater than the maximum size for an IP packet.

3040 400026  TCP NULL flags  Attack  Triggers when a single TCP packet with none of the SYN, FIN, ACK, or RST flags set has been sent to a specific host.

3041 400027  TCP SYN+FIN flags  Attack  Triggers when a single TCP packet with the SYN and FIN flags are set and is sent to a specific host.

Table 48-1  Signature IDs and System Message Numbers (continued)
### Table 48-1 Signature IDs and System Message Numbers (continued)

<table>
<thead>
<tr>
<th>Signature ID</th>
<th>Message Number</th>
<th>Signature Title</th>
<th>Signature Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>6051</td>
<td>400035</td>
<td>DNS Zone Transfer</td>
<td>Informational</td>
<td>Triggers on normal DNS zone transfers, in which the source port is 53.</td>
</tr>
<tr>
<td>6052</td>
<td>400036</td>
<td>DNS Zone Transfer from High Port</td>
<td>Informational</td>
<td>Triggers on an illegitimate DNS zone transfer, in which the source port is not equal to 53.</td>
</tr>
<tr>
<td>6053</td>
<td>400037</td>
<td>DNS Request for All Records</td>
<td>Informational</td>
<td>Triggers on a DNS request for all records.</td>
</tr>
<tr>
<td>6100</td>
<td>400038</td>
<td>RPC Port Registration</td>
<td>Informational</td>
<td>Triggers when attempts are made to register new RPC services on a target host.</td>
</tr>
<tr>
<td>6101</td>
<td>400039</td>
<td>RPC Port Unregistration</td>
<td>Informational</td>
<td>Triggers when attempts are made to unregister existing RPC services on a target host.</td>
</tr>
<tr>
<td>6102</td>
<td>400040</td>
<td>RPC Dump</td>
<td>Informational</td>
<td>Triggers when an RPC dump request is issued to a target host.</td>
</tr>
<tr>
<td>6103</td>
<td>400041</td>
<td>Proxied RPC Request</td>
<td>Attack</td>
<td>Triggers when a proxied RPC request is sent to the portmapper of a target host.</td>
</tr>
<tr>
<td>6150</td>
<td>400042</td>
<td>ypserv (YP server daemon) Portmap Request</td>
<td>Informational</td>
<td>Triggers when a request is made to the portmapper for the YP server daemon (ypserv) port.</td>
</tr>
<tr>
<td>6151</td>
<td>400043</td>
<td>ypbind (YP bind daemon) Portmap Request</td>
<td>Informational</td>
<td>Triggers when a request is made to the portmapper for the YP bind daemon (ypbind) port.</td>
</tr>
<tr>
<td>6152</td>
<td>400044</td>
<td>yppasswdd (YP password daemon) Portmap Request</td>
<td>Informational</td>
<td>Triggers when a request is made to the portmapper for the YP password daemon (yppasswdd) port.</td>
</tr>
<tr>
<td>6153</td>
<td>400045</td>
<td>ypupdated (YP update daemon) Portmap Request</td>
<td>Informational</td>
<td>Triggers when a request is made to the portmapper for the YP update daemon (ypupdated) port.</td>
</tr>
<tr>
<td>6154</td>
<td>400046</td>
<td>ypxfrd (YP transfer daemon) Portmap Request</td>
<td>Informational</td>
<td>Triggers when a request is made to the portmapper for the YP transfer daemon (ypxfrd) port.</td>
</tr>
<tr>
<td>6155</td>
<td>400047</td>
<td>mountd (mount daemon) Portmap Request</td>
<td>Informational</td>
<td>Triggers when a request is made to the portmapper for the mount daemon (mountd) port.</td>
</tr>
<tr>
<td>6175</td>
<td>400048</td>
<td>rexd (remote execution daemon) Portmap Request</td>
<td>Informational</td>
<td>Triggers when a request is made to the portmapper for the remote execution daemon (rexd) port.</td>
</tr>
</tbody>
</table>
### Table 48-1 Signature IDs and System Message Numbers (continued)

<table>
<thead>
<tr>
<th>Signature ID</th>
<th>Message Number</th>
<th>Signature Title</th>
<th>Signature Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>6180</td>
<td>400049</td>
<td>rexd (remote execution daemon) Attempt</td>
<td>Informational</td>
<td>Triggers when a call to the rexd program is made. The remote execution daemon is the server responsible for remote program execution. This may be indicative of an attempt to gain unauthorized access to system resources.</td>
</tr>
<tr>
<td>6190</td>
<td>400050</td>
<td>statd Buffer Overflow</td>
<td>Attack</td>
<td>Triggers when a large statd request is sent. This could be an attempt to overflow a buffer and gain access to system resources.</td>
</tr>
</tbody>
</table>
PART 13

Configuring High Availability
Information About High Availability

This chapter provides an overview of the failover features that enable you to achieve high availability on the Cisco 5500 series ASASMs. For information about configuring high availability, see Chapter 51, “Configuring Active/Active Failover” or Chapter 50, “Configuring Active/Standby Failover.”

This chapter includes the following sections:

- Introduction to Failover and High Availability, page 49-1
- Failover System Requirements, page 49-2
- Failover and Stateful Failover Links, page 49-2
- Active/Active and Active/Standby Failover, page 49-7
- Stateless (Regular) and Stateful Failover, page 49-8
- Intra- and Inter-Chassis Module Placement for the ASA Services Module, page 49-10
- Transparent Firewall Mode Requirements, page 49-13
- Auto Update Server Support in Failover Configurations, page 49-14
- Failover Health Monitoring, page 49-16
- Failover Times, page 49-18
- Failover Messages, page 49-18

Introduction to Failover and High Availability

Configuring high availability requires two identical ASASMs connected to each other through a dedicated failover link and, optionally, a Stateful Failover link. The health of the active interfaces and units is monitored to determine if specific failover conditions are met. If those conditions are met, failover occurs.

The ASASM supports two failover configurations, Active/Active failover and Active/Standby failover. Each failover configuration has its own method for determining and performing failover.

With Active/Active failover, both units can pass network traffic. This also lets you configure traffic sharing on your network. Active/Active failover is available only on units running in multiple context mode.

With Active/Standby failover, only one unit passes traffic while the other unit waits in a standby state. Active/Standby failover is available on units running in either single or multiple context mode.

Both failover configurations support stateful or stateless (regular) failover.
Failover System Requirements

This section describes the hardware, software, and license requirements for ASASMs in a failover configuration.

This section includes the following topics:
- Software Requirements, page 49-2
- License Requirements, page 49-2

Software Requirements

The two units in a failover configuration must be in the same operating modes (single or multiple context). They must have the same major (first number) and minor (second number) software version. However, you can use different versions of the software during an upgrade process; for example, you can upgrade one unit from Version 8.3(1) to Version 8.3(2) and have failover remain active. We recommend upgrading both units to the same version to ensure long-term compatibility.

See the “Performing Zero Downtime Upgrades for Failover Pairs” section on page 56-6 for more information about upgrading the software on a failover pair.

License Requirements

The two units in a failover configuration do not need to have identical licenses; the licenses combine to make a failover cluster license. See the “Failover Licenses” section on page 4-6 for more information.

Failover and Stateful Failover Links

This section describes the failover and the Stateful Failover links, which are dedicated connections between the two units in a failover configuration. This section includes the following topics:
- Failover Link, page 49-2
- Stateful Failover Link, page 49-3
- Avoiding Interrupted Failover Links, page 49-4

Failover Link

The two units in a failover pair constantly communicate over a failover link to determine the operating status of each unit. The following information is communicated over the failover link:
- The unit state (active or standby)
- Hello messages (keep-alives)
- Network link status
- MAC address exchange
- Configuration replication and synchronization
Caution

All information sent over the failover and Stateful Failover links is sent in clear text unless you secure the communication with a failover key.

You can use any unused interface on the device as the failover link; however, you cannot specify an interface that is currently configured with a name. The failover link interface is not configured as a normal networking interface; it exists for failover communication only. This interface should only be used for the failover link (and optionally for the Stateful Failover link).

Connect the failover link in one of the following two ways:

- Using a switch, with no other device on the same network segment (broadcast domain or VLAN) as the failover interfaces of the ASASM.
- Using a crossover Ethernet cable to connect the appliances directly, without the need for an external switch.

Note

When you use a crossover cable for the failover link, if the interface fails, the link is brought down on both peers. This condition may hamper troubleshooting efforts because you cannot easily determine which interface failed and caused the link to come down.

Although you can configure failover and failover state links on a port channel link, this port channel cannot be shared with other firewall traffic.

Stateful Failover Link

To use Stateful Failover, you must configure a Stateful Failover link to pass all state information. You have three options for configuring a Stateful Failover link:

- You can use a dedicated Ethernet interface for the Stateful Failover link.
- You can share the failover link.
- You can share a regular data interface, such as the inside interface. However, this option is not recommended.

Connect a dedicated state link in one of the following two ways:

- Using a switch, with no other device on the same network segment (broadcast domain or VLAN) as the failover interfaces of the ASASM.
- Using a crossover Ethernet cable to connect the appliances directly, without the need for an external switch.

Note

When you use a crossover cable for the state link, if the interface fails, the link is brought down on both peers. This condition may hamper troubleshooting efforts because you cannot easily determine which interface failed and caused the link to come down.

Enable the PortFast option on Cisco switch ports that connect directly to the ASASM.

If you use a data interface as the Stateful Failover link, you receive the following warning when you specify that interface as the Stateful Failover link:

******** WARNING ***** WARNING ******* WARNING ***** WARNING ******* WARNING
Sharing Stateful failover interface with regular data interface is not a recommended configuration due to performance and security concerns.

******** WARNING ***** WARNING ******** WARNING ******** WARNING ********

Sharing a data interface with the Stateful Failover interface can leave you vulnerable to replay attacks. Additionally, large amounts of Stateful Failover traffic may be sent on the interface, causing performance problems on that network segment.

**Note**
Using a data interface as the Stateful Failover interface is supported in single context, routed mode only.

In multiple context mode, the Stateful Failover link resides in the system context. This interface and the failover interface are the only interfaces in the system context. All other interfaces are allocated to and configured from within security contexts.

**Note**
The IP address and MAC address for the Stateful Failover link does not change at failover unless the Stateful Failover link is configured on a regular data interface.

**Caution**
All information sent over the failover and Stateful Failover links is sent in clear text unless you secure the communication with a failover key.

**Failover Interface Speed for Stateful Links**

If you use the failover link as the Stateful Failover link, you should use the fastest Ethernet interface available. If you experience performance problems on that interface, consider dedicating a separate interface for the Stateful Failover interface.

For optimum performance when using long distance failover, the latency for the failover link should be less than 10 milliseconds and no more than 250 milliseconds. If latency is more than 10 milliseconds, some performance degradation occurs due to retransmission of failover messages.

The ASASM supports sharing of failover heartbeat and stateful link, but we recommend using a separate heartbeat link on systems with high Stateful Failover traffic.

**Avoiding Interrupted Failover Links**

Because the uses failover interfaces to transport messages between primary and secondary units, if a failover interface is down (that is, the physical link is down or the switch used to connect the interface is down), then the ASASM failover operation is affected until the health of the failover interface is restored.

In the event that all communication is cut off between the units in a failover pair, both units go into the active state, which is expected behavior. When communication is restored and the two active units resume communication through the failover link or through any monitored interface, the primary unit remains active, and the secondary unit immediately returns to the standby state. This relationship is established regardless of the health of the primary unit.

Because of this behavior, stateful flows that were passed properly by the secondary active unit during the network split are now interrupted. To avoid this interruption, failover links and data interfaces should travel through different paths to decrease the chance that all links fail at the same time. In the event that only one failover link is down, the ASASM takes a sample of the interface health, exchanges this
information with its peer through the data interface, and performs a switchover if the active unit has a greater number of down interfaces. Subsequently, the failover operation is suspended until the health of the failover link is restored.

Depending upon their network topologies, several primary/secondary failure scenarios exist in ASASM failover pairs, as shown in the following scenarios.

**Scenario 1—Not Recommended**

If a single switch or a set of switches are used to connect both failover and data interfaces between two ASASMs, then when a switch or inter-switch-link is down, both ASASMs become active. Therefore, the following two connection methods shown in Figure 49-1 and Figure 49-2 are NOT recommended.

**Scenario 2—Recommended**

To make the ASA failover pair resistant to failover interface failure, we recommend that failover interfaces NOT use the same switch as the data interfaces, as shown in the preceding connections. Instead, use a different switch or use a direct cable to connect two ASASM failover interfaces, as shown in Figure 49-3 and Figure 49-4.
Scenario 3—Recommended
If the ASASM data interfaces are connected to more than one set of switches, then a failover interface can be connected to one of the switches, preferably the switch on the secure side of network, as shown in Figure 49-5.

Scenario 4—Recommended
The most reliable failover configurations use a redundant interface on the failover interface, as shown in Figure 49-6 and Figure 49-7.
Active/Active and Active/Standby Failover

Two types of failover configurations are supported by the ASASM: Active/Standby and Active/Active. In Active/Standby failover, one unit is the active unit. It passes traffic. The standby unit does not actively pass traffic. When a failover occurs, the active unit fails over to the standby unit, which then becomes active. You can use Active/Standby failover for ASASMs in single or multiple context mode, although it is most commonly used for ASASMs in single context mode.

Active/Active failover is only available to ASASMs in multiple context mode. In an Active/Active failover configuration, both ASASMs can pass network traffic. In Active/Active failover, you divide the security contexts on the ASASM into failover groups. A failover group is simply a logical group of one or more security contexts. Each group is assigned to be active on a specific ASASM in the failover pair. When a failover occurs, it occurs at the failover group level.

For more detailed information about each type of failover, refer to the following information:
- Chapter 50, “Configuring Active/Standby Failover”
- Chapter 51, “Configuring Active/Active Failover”

Determining Which Type of Failover to Use

The type of failover you choose depends upon your ASASM configuration and how you plan to use the ASASMs.

If you are running the ASASM in single mode, then you can use only Active/Standby failover. Active/Active failover is only available to ASASMs running in multiple context mode.
If you are running the ASASM in multiple context mode, then you can configure either Active/Active failover or Active/Standby failover.

- To allow both members of the failover pair to share the traffic, use Active/Active failover. Do not exceed 50% load on each device.
- If you do not want to share the traffic in this way, use Active/Standby or Active/Active failover.

Table 49-1 provides a comparison of some of the features supported by each type of failover configuration.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Active/Active</th>
<th>Active/Standby</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single Context Mode</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Multiple Context Mode</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Traffic Sharing Network Configurations</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Unit Failover</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Failover of Groups of Contexts</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Failover of Individual Contexts</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>

Note: If single signon on the Catalyst 6500 Series switch takes 1 or more seconds, failover occurs on the ASA SM because the unit polltime is less than this value.

Stateless (Regular) and Stateful Failover

The ASASM supports two types of failover, regular and stateful. This section includes the following topics:

- Stateless (Regular) Failover, page 49-8
- Stateful Failover, page 49-8

Stateless (Regular) Failover

When a failover occurs, all active connections are dropped. Clients need to reestablish connections when the new active unit takes over.

Stateful Failover

When Stateful Failover is enabled, the active unit continually passes per-connection state information to the standby unit. After a failover occurs, the same connection information is available at the new active unit. Supported end-user applications are not required to reconnect to keep the same communication session.

In Version 8.4 and later, Stateful Failover participates in dynamic routing protocols, like OSPF and EIGRP, so routes that are learned through dynamic routing protocols on the active unit are maintained in a Routing Information Base (RIB) table on the standby unit. Upon a failover event, packets travel
Stateless (Regular) and Stateful Failover

normally with minimal disruption to traffic because the Active secondary ASASM initially has rules that mirror the primary ASASM. Immediately after failover, the re-convergence timer starts on the newly Active unit. Then the epoch number for the RIB table increments. During re-convergence, OSPF and EIGRP routes become updated with a new epoch number. Once the timer is expired, stale route entries (determined by the epoch number) are removed from the table. The RIB then contains the newest routing protocol forwarding information on the newly Active unit.

Table 49-2 list the state information that is and is not passed to the standby unit when Stateful Failover is enabled.

Table 49-2 State Information

<table>
<thead>
<tr>
<th>State Information Passed to Standby Unit</th>
<th>State Information Not Passed to Standby Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAT translation table</td>
<td>The HTTP connection table (unless HTTP replication is enabled).</td>
</tr>
<tr>
<td>TCP connection states</td>
<td>The user authentication (uauth) table.</td>
</tr>
<tr>
<td></td>
<td>Inspected protocols are subject to advanced TCP-state tracking, and the TCP state of these connections is not automatically replicated. While these connections are replicated to the standby unit, there is a best-effort attempt to re-establish a TCP state.</td>
</tr>
<tr>
<td>UDP connection states</td>
<td>DHCP server address leases.</td>
</tr>
<tr>
<td>The ARP table</td>
<td>State information for modules.</td>
</tr>
<tr>
<td>The Layer 2 bridge table (when running in transparent firewall mode)</td>
<td>Stateful Failover for phone proxy. When the active unit goes down, the call fails, media stops flowing, and the phone should unregister from the failed unit and reregister with the active unit. The call must be re-established.</td>
</tr>
<tr>
<td>The HTTP connection states (if HTTP replication is enabled)</td>
<td>—</td>
</tr>
<tr>
<td>GTP PDP connection database</td>
<td>—</td>
</tr>
<tr>
<td>SIP signalling sessions</td>
<td>—</td>
</tr>
<tr>
<td>ICMP connection state</td>
<td>By default, the ASA does not replicate the ICMP connection state in failover. ICMP connection replication is enabled only if the respective interface is assigned to an asymmetric routing group.</td>
</tr>
</tbody>
</table>

Note

If failover occurs during an active Cisco IP SoftPhone session, the call remains active because the call session state information is replicated to the standby unit. When the call is terminated, the IP SoftPhone client loses connection with the Cisco CallManager. This occurs because there is no session information for the CTIQBE hangup message on the standby unit. When the IP SoftPhone client does not receive a response back from the Call Manager within a certain time period, it considers the CallManager unreachable and unregisters itself.
Intra- and Inter-Chassis Module Placement for the ASA Services Module

You can place the primary and secondary ASASMs within the same switch or in two separate switches. The following sections describe each option:

- Intra-Chassis Failover, page 49-10
- Inter-Chassis Failover, page 49-10

Intra-Chassis Failover

If you install the secondary ASASM in the same switch as the primary ASASM, you protect against module-level failure. To protect against switch-level failure, as well as module-level failure, see the “Inter-Chassis Failover” section on page 49-10.

Even though both ASASMs are assigned the same VLANs, only the active module takes part in networking. The standby module does not pass any traffic.

Figure 49-8 shows a typical intra-switch configuration.

Inter-Chassis Failover

To protect against switch-level failure, you can install the secondary ASASM in a separate switch. The ASASM does not coordinate failover directly with the switch, but it works harmoniously with the switch failover operation. See the switch documentation to configure failover for the switch.
To accommodate the failover communications between ASASMs, we recommend that you configure a trunk port between the two switches that carries the failover and state VLANs. The trunk ensures that failover communication between the two units is subject to minimal failure risk.

For other VLANs, you must ensure that both switches have access to all firewall VLANs, and that monitored VLANs can successfully pass hello packets between both switches.

Figure 49-9 shows a typical switch and ASASM redundancy configuration. The trunk between the two switches carries the failover ASASM VLANs (VLANs 10 and 11).

**Note** ASASM failover is independent of the switch failover operation; however, ASASM works in any switch failover scenario.

Figure 49-9  Normal Operation
If the primary ASASM fails, then the secondary ASASM becomes active and successfully passes the firewall VLANs (Figure 49-10).

*Figure 49-10  ASASM Failure*
If the entire switch fails, as well as the ASASM (such as in a power failure), then both the switch and the ASASM fail over to their secondary units (Figure 49-11).

**Figure 49-11 Switch Failure**

---

### Transparent Firewall Mode Requirements

When the active unit fails over to the standby unit, the connected switch port running Spanning Tree Protocol (STP) can go into a blocking state for 30 to 50 seconds when it senses the topology change. To avoid traffic loss while the port is in a blocking state, you can configure one of the following workarounds depending on the switch port mode:

- **Access mode**—Enable the STP PortFast feature on the switch:

  ```
  interface interface_id
  spanning-tree portfast
  ```

  The PortFast feature immediately transitions the port into STP forwarding mode upon linkup. The port still participates in STP. So if the port is to be a part of the loop, the port eventually transitions into STP blocking mode.

- **Trunk mode**—Block BPDUs on the ASASM on both the inside and outside interfaces:

  ```
  access-list id ethertype deny bpdu
  access-group id in interface inside_name
  access-group id in interface outside_name
  ```

---
Blocking BPDUs disables STP on the switch. Be sure not to have any loops involving the ASASM in your network layout.

If neither of the above options are possible, then you can use one of the following less desirable workarounds that impacts failover functionality or STP stability:

- Disable failover interface monitoring.
- Increase failover interface holdtime to a high value that will allow STP to converge before the ASASMs fail over.
- Decrease STP timers to allow STP to converge faster than the failover interface holdtime.

**Auto Update Server Support in Failover Configurations**

You can use the Auto Update Server to deploy software images and configuration files to ASASMs in an Active/Standby failover configuration. To enable Auto Update on an Active/Standby failover configuration, enter the Auto Update Server configuration on the primary unit in the failover pair. See the “Configuring Auto Update Support” section on page 56-16, for more information.

The following restrictions and behaviors apply to Auto Update Server support in failover configurations:

- Only single mode, Active/Standby configurations are supported.
- When loading a new platform software image, the failover pair stops passing traffic.
- When using LAN-based failover, new configurations must not change the failover link configuration. If they do, communication between the units will fail.
- Only the primary unit will perform the call home to the Auto Update Server. The primary unit must be in the active state to call home. If it is not, the ASASM automatically fails over to the primary unit.
- Only the primary unit downloads the software image or configuration file. The software image or configuration is then copied to the secondary unit.
- The interface MAC address and hardware-serial ID is from the primary unit.
- The configuration file stored on the Auto Update Server or HTTP server is for the primary unit only.

**Auto Update Process Overview**

The following is an overview of the Auto Update process in failover configurations. This process assumes that failover is enabled and operational. The Auto Update process cannot occur if the units are synchronizing configurations, if the standby unit is in the failed state for any reason other than SSM card failure, or if the failover link is down.

1. Both units exchange the platform and ASDM software checksum and version information.
2. The primary unit contacts the Auto Update Server. If the primary unit is not in the active state, the ASASM first fails over to the primary unit and then contacts the Auto Update Server.
3. The Auto Update Server replies with software checksum and URL information.
4. If the primary unit determines that the platform image file needs to be updated for either the active or standby unit, the following occurs:
   a. The primary unit retrieves the appropriate files from the HTTP server using the URL from the Auto Update Server.
b. The primary unit copies the image to the standby unit and then updates the image on itself.

c. If both units have a new image, the secondary (standby) unit is reloaded first.
   - If a hitless upgrade can be performed when the secondary unit boots, then the secondary unit becomes the active unit and the primary unit reloads. The primary unit becomes the active unit when it has finished loading.
   - If a hitless upgrade cannot be performed when the standby unit boots, then both units reload at the same time.

d. If only the secondary (standby) unit has a new image, then only the secondary unit reloads. The primary unit waits until the secondary unit finishes reloading.

e. If only the primary (active) unit has a new image, the secondary unit becomes the active unit, and the primary unit reloads.

f. The update process starts again at Step 1.

5. If the ASASM determines that the ASDM file needs to be updated for either the primary or secondary unit, the following occurs:

   a. The primary unit retrieves the ASDM image file from the HTTP server using the URL provided by the Auto Update Server.

   b. The primary unit copies the ASDM image to the standby unit, if needed.

   c. The primary unit updates the ASDM image on itself.

   d. The update process starts again at Step 1.

6. If the primary unit determines that the configuration needs to be updated, the following occurs:

   a. The primary unit retrieves the configuration file from the using the specified URL.

   b. The new configuration replaces the old configuration on both units simultaneously.

   c. The update process begins again at Step 1.

7. If the checksums match for all image and configuration files, no updates are required. The process ends until the next poll time.

### Monitoring the Auto Update Process

You can use the `debug auto-update client` or `debug fover cmd-exe` commands to display the actions performed during the Auto Update process. The following is sample output from the `debug auto-update client` command.

```
Auto-update client: Sent DeviceDetails to /cgi-bin/dda.pl of server 192.168.0.21
Auto-update client: Processing UpdateInfo from server 192.168.0.21
  Component: asdm, URL: http://192.168.0.21/asdm.bint, checksum:
  0x94bced0261cc992ae710faf8d244cf32
  Component: config, URL: http://192.168.0.21/config-rms.xml, checksum:
  0x67358553572688a805a155af312f6898
  Component: image, URL: http://192.168.0.21/cdisk73.bin, checksum:
  0x6d091b43ce96243e29a62f2330139419
Auto-update client: need to update img, act: yes, stby yes
name
ciscoasa(config)# Auto-update client: update img on stby unit...
  auto-update: Fover copyfile, seq = 4 type = 1, pseq = 1, len = 1024
  auto-update: Fover copyfile, seq = 4 type = 1, pseq = 501, len = 1024
  auto-update: Fover copyfile, seq = 4 type = 1, pseq = 1001, len = 1024
  auto-update: Fover copyfile, seq = 4 type = 1, pseq = 1501, len = 1024
  auto-update: Fover copyfile, seq = 4 type = 1, pseq = 2001, len = 1024
```
Failover Health Monitoring

The ASASM monitors each unit for overall health and for interface health. See the following sections for more information about how the ASASM performs tests to determine the state of each unit:

- **Unit Health Monitoring, page 49-17**
- **Interface Monitoring, page 49-17**
Unit Health Monitoring

The ASASM determines the health of the other unit by monitoring the failover link. When a unit does not receive three consecutive hello messages on the failover link, the unit sends interface hello messages on each interface, including the failover interface, to validate whether or not the peer interface is responsive. The action that the ASASM takes depends upon the response from the other unit. See the following possible actions:

- If the ASASM receives a response on the failover interface, then it does not fail over.
- If the ASASM does not receive a response on the failover link, but it does receive a response on another interface, then the unit does not failover. The failover link is marked as failed. You should restore the failover link as soon as possible because the unit cannot fail over to the standby while the failover link is down.
- If the ASASM does not receive a response on any interface, then the standby unit switches to active mode and classifies the other unit as failed.

You can configure the frequency of the hello messages and the hold time before failover occurs. A faster poll time and shorter hold time speed the detection of unit failures and make failover occur more quickly, but it can also cause “false” failures due to network congestion delaying the keepalive packets.

Interface Monitoring

You can monitor up to 250 interfaces divided between all contexts. You should monitor important interfaces. For example, you might configure one context to monitor a shared interface. (Because the interface is shared, all contexts benefit from the monitoring.)

When a unit does not receive hello messages on a monitored interface for half of the configured hold time, it runs the following tests:

1. Link Up/Down test—A test of the interface status. If the Link Up/Down test indicates that the interface is operational, then the ASASM performs network tests. The purpose of these tests is to generate network traffic to determine which (if either) unit has failed. At the start of each test, each unit clears its received packet count for its interfaces. At the conclusion of each test, each unit looks to see if it has received any traffic. If it has, the interface is considered operational. If one unit receives traffic for a test and the other unit does not, the unit that received no traffic is considered failed. If neither unit has received traffic, then the next test is used.

2. Network Activity test—A received network activity test. The unit counts all received packets for up to 5 seconds. If any packets are received at any time during this interval, the interface is considered operational and testing stops. If no traffic is received, the ARP test begins.

3. ARP test—A reading of the unit ARP cache for the 2 most recently acquired entries. One at a time, the unit sends ARP requests to these machines, attempting to stimulate network traffic. After each request, the unit counts all received traffic for up to 5 seconds. If traffic is received, the interface is considered operational. If no traffic is received, an ARP request is sent to the next machine. If at the end of the list no traffic has been received, the ping test begins.

4. Broadcast Ping test—A ping test that consists of sending out a broadcast ping request. The unit then counts all received packets for up to 5 seconds. If any packets are received at any time during this interval, the interface is considered operational and testing stops.

If an interface has IPv4 and IPv6 addresses configured on it, the ASASM uses the IPv4 addresses to perform the health monitoring.
If an interface has only IPv6 addresses configured on it, then the ASASM uses IPv6 neighbor discovery instead of ARP to perform the health monitoring tests. For the broadcast ping test, the ASASM uses the IPv6 all nodes address (FE02::1).

If all network tests fail for an interface, but this interface on the other unit continues to successfully pass traffic, then the interface is considered to be failed. If the threshold for failed interfaces is met, then a failover occurs. If the other unit interface also fails all the network tests, then both interfaces go into the “Unknown” state and do not count towards the failover limit.

An interface becomes operational again if it receives any traffic. A failed ASASM returns to standby mode if the interface failure threshold is no longer met.

Note

If a failed unit does not recover and you believe it should not be failed, you can reset the state by entering the `failover reset` command. If the failover condition persists, however, the unit will fail again.

## Failover Times

Table 49-3 shows the minimum, default, and maximum failover times.

<table>
<thead>
<tr>
<th>Failover Condition</th>
<th>Minimum</th>
<th>Default</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active unit loses power or stops normal operation.</td>
<td>800 milliseconds</td>
<td>15 seconds</td>
<td>45 seconds</td>
</tr>
<tr>
<td>Active unit main board interface link down.</td>
<td>500 milliseconds</td>
<td>5 seconds</td>
<td>15 seconds</td>
</tr>
<tr>
<td>Active unit 4GE module interface link down.</td>
<td>2 seconds</td>
<td>5 seconds</td>
<td>15 seconds</td>
</tr>
<tr>
<td>Active unit IPS or CSC module fails.</td>
<td>2 seconds</td>
<td>2 seconds</td>
<td>2 seconds</td>
</tr>
<tr>
<td>Active unit interface up, but connection problem causes interface testing.</td>
<td>5 seconds</td>
<td>25 seconds</td>
<td>75 seconds</td>
</tr>
</tbody>
</table>

## Failover Messages

When a failover occurs, both ASASMs send out system messages. This section includes the following topics:

- Failover System Messages, page 49-18
- Debug Messages, page 49-19
- SNMP, page 49-19

## Failover System Messages

The ASASM issues a number of system messages related to failover at priority level 2, which indicates a critical condition. To view these messages, see the `syslog messages guide` guide. To enable logging, see Chapter 52, “Configuring Logging.”
Note

During switchover, failover logically shuts down and then bring up interfaces, generating syslog messages 411001 and 411002. This is normal activity.

Debug Messages

To see debug messages, enter the `debug fover` command. See the command reference for more information.

Note

Because debugging output is assigned high priority in the CPU process, it can drastically affect system performance. For this reason, use the `debug fover` commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco TAC.

SNMP

To receive SNMP syslog traps for failover, configure the SNMP agent to send SNMP traps to SNMP management stations, define a syslog host, and compile the Cisco syslog MIB into your SNMP management station. See Chapter 54, “Configuring SNMP” for more information.
Failover Messages
Configuring Active/Standby Failover

This chapter describes how to configure Active/Standby failover and includes the following sections:

- Information About Active/Standby Failover, page 50-1
- Licensing Requirements for Active/Standby Failover, page 50-6
- Prerequisites for Active/Standby Failover, page 50-6
- Guidelines and Limitations, page 50-6
- Configuring Active/Standby Failover, page 50-7
- Controlling Failover, page 50-16
- Monitoring Active/Standby Failover, page 50-18
- Feature History for Active/Standby Failover, page 50-18

Information About Active/Standby Failover

This section describes Active/Standby failover and includes the following topics:

- Active/Standby Failover Overview, page 50-1
- Primary/Secondary Status and Active/Standby Status, page 50-2
- Device Initialization and Configuration Synchronization, page 50-2
- Command Replication, page 50-3
- Failover Triggers, page 50-4
- Failover Actions, page 50-4

Active/Standby Failover Overview

Active/Standby failover enables you to use a standby ASASM to take over the functionality of a failed unit. When the active unit fails, it changes to the standby state while the standby unit changes to the active state. The unit that becomes active assumes the IP addresses (or, for transparent firewall, the management IP address) and MAC addresses of the failed unit and begins passing traffic. The unit that is now in standby state takes over the standby IP addresses and MAC addresses. Because network devices see no change in the MAC to IP address pairing, no ARP entries change or time out anywhere on the network.
For multiple context mode, the ASASM can fail over the entire unit (including all contexts) but cannot fail over individual contexts separately.

Primary/Secondary Status and Active/Standby Status

The main differences between the two units in a failover pair are related to which unit is active and which unit is standby, namely which IP addresses to use and which unit actively passes traffic.

However, a few differences exist between the units based on which unit is primary (as specified in the configuration) and which unit is secondary:

- The primary unit always becomes the active unit if both units start up at the same time (and are of equal operational health).
- The primary unit MAC addresses are always coupled with the active IP addresses. The exception to this rule occurs when the secondary unit is active and cannot obtain the primary unit MAC addresses over the failover link. In this case, the secondary unit MAC addresses are used.

Device Initialization and Configuration Synchronization

Configuration synchronization occurs when one or both devices in the failover pair boot. Configurations are always synchronized from the active unit to the standby unit. When the standby unit completes its initial startup, it clears its running configuration (except for the failover commands needed to communicate with the active unit), and the active unit sends its entire configuration to the standby unit.

The active unit is determined by the following:

- If a unit boots and detects a peer already running as active, it becomes the standby unit.
- If a unit boots and does not detect a peer, it becomes the active unit.
- If both units boot simultaneously, then the primary unit becomes the active unit, and the secondary unit becomes the standby unit.

If the secondary unit boots without detecting the primary unit, it becomes the active unit. It uses its own MAC addresses for the active IP addresses. However, when the primary unit becomes available, the secondary unit changes the MAC addresses to those of the primary unit, which can cause an interruption in your network traffic. To avoid this, configure the failover pair with virtual MAC addresses. See the “Configuring Virtual MAC Addresses” section on page 50-15 for more information.

When the replication starts, the ASASM console on the active unit displays the message “Beginning configuration replication: Sending to mate,” and when it is complete, the ASASM displays the message “End Configuration Replication to mate.” During replication, commands entered on the active unit may not replicate properly to the standby unit, and commands entered on the standby unit may be overwritten by the configuration being replicated from the active unit. Avoid entering commands on either unit in the failover pair during the configuration replication process. Depending upon the size of the configuration, replication can take from a few seconds to several minutes.

The `crypto ca server` command and related sub commands are not synchronized to the failover peer.
On the standby unit, the configuration exists only in running memory. To save the configuration to flash memory after synchronization, do the following:

- For single context mode, enter the **write memory** command on the active unit. The command is replicated to the standby unit, which proceeds to write its configuration to flash memory.

- For multiple context mode, enter the **write memory all** command on the active unit from the system execution space. The command is replicated to the standby unit, which proceeds to write its configuration to flash memory. Using the **all** keyword with this command causes the system and all context configurations to be saved.

**Note**

Startup configurations saved on external servers are accessible from either unit over the network and do not need to be saved separately for each unit. Alternatively, you can copy the contexts on disk from the active unit to an external server, and then copy them to disk on the standby unit, where they become available when the unit reloads.

## Command Replication

Command replication always flows from the active unit to the standby unit. As commands are entered on the active unit, they are sent across the failover link to the standby unit. You do not have to save the active configuration to flash memory to replicate the commands.

*Table 50-1* lists the commands that are and are not replicated to the standby unit.

<table>
<thead>
<tr>
<th>Command Replicated to the Standby Unit</th>
<th>Commands Not Replicated to the Standby Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>All configuration commands except for <strong>mode</strong>, <strong>firewall</strong>, and <strong>failover lan unit</strong></td>
<td>All forms of the <strong>copy</strong> command except for <strong>copy running-config startup-config</strong></td>
</tr>
<tr>
<td><strong>copy running-config startup-config</strong></td>
<td>all forms of the <strong>write</strong> command except for <strong>write memory</strong></td>
</tr>
<tr>
<td><strong>delete</strong></td>
<td><strong>crypto ca server</strong> and associated sub commands</td>
</tr>
<tr>
<td><strong>mkdir</strong></td>
<td><strong>debug</strong></td>
</tr>
<tr>
<td><strong>rename</strong></td>
<td><strong>failover lan unit</strong></td>
</tr>
<tr>
<td><strong>rmdir</strong></td>
<td><strong>firewall</strong></td>
</tr>
<tr>
<td><strong>write memory</strong></td>
<td><strong>mode</strong></td>
</tr>
<tr>
<td><strong>—</strong></td>
<td><strong>show</strong></td>
</tr>
<tr>
<td><strong>—</strong></td>
<td><strong>terminal pager</strong> and <strong>pager</strong></td>
</tr>
</tbody>
</table>

**Note**

Changes made on the standby unit are not replicated to the active unit. If you enter a command on the standby unit, the ASASM displays the message **** WARNING **** Configuration Replication is NOT performed from Standby unit to Active unit. Configurations are no longer synchronized. This message appears even when you enter many commands that do not affect the configuration.

If you enter the **write standby** command on the active unit, the standby unit clears its running configuration (except for the failover commands used to communicate with the active unit), and the active unit sends its entire configuration to the standby unit.
For multiple context mode, when you enter the `write standby` command in the system execution space, all contexts are replicated. If you enter the `write standby` command within a context, the command replicates only the context configuration.

Replicated commands are stored in the running configuration.

---

**Note**

Standby Failover does not replicate the following files and configuration components:

- AnyConnect images
- CSD images
- ASA images
- AnyConnect profiles
- Local Certificate Authorities (CAs)
- ASDM images

---

To save the replicated commands to the flash memory on the standby unit, standby unit, do the following:

- For single context mode, enter the `copy running-config startup-config` command on the active unit. The command is replicated to the standby unit, which proceeds to write its configuration to flash memory.
- For multiple context mode, enter the `copy running-config startup-config` command on the active unit from the system execution space and within each context on disk. The command is replicated to the standby unit, which proceeds to write its configuration to flash memory. Contexts with startup configurations on external servers are accessible from either unit over the network and do not need to be saved separately for each unit. Alternatively, you can copy the contexts on disk from the active unit to an external server, and then copy them to disk on the standby unit.

**Failover Triggers**

The unit can fail if one of the following events occurs:

- The unit has a hardware failure or a power failure.
- The unit has a software failure.
- Too many monitored interfaces fail.
- You force a failover. (See the “Forcing Failover” section on page 50-16.)

**Failover Actions**

In Active/Standby failover, failover occurs on a unit basis. Even on systems running in multiple context mode, you cannot fail over individual or groups of contexts.
Table 50-2 shows the failover action for each failure event. For each failure event, the table shows the failover policy (failover or no failover), the action taken by the active unit, the action taken by the standby unit, and any special notes about the failover condition and actions.

**Table 50-2  Failover Behavior**

<table>
<thead>
<tr>
<th>Failure Event</th>
<th>Policy</th>
<th>Active Action</th>
<th>Standby Action</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active unit failed (power or hardware)</td>
<td>Failover</td>
<td>n/a</td>
<td>Become active</td>
<td>Mark active as failed</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>No hello messages are received on any monitored interface or the failover link.</td>
</tr>
<tr>
<td>Formerly active unit recovers</td>
<td>No failover</td>
<td>Become standby</td>
<td>No action</td>
<td>None.</td>
</tr>
<tr>
<td>Standby unit failed (power or hardware)</td>
<td>No failover</td>
<td>Mark standby as failed</td>
<td>n/a</td>
<td>When the standby unit is marked as failed, then the active unit does not attempt to fail over, even if the interface failure threshold is surpassed.</td>
</tr>
<tr>
<td>Failover link failed during operation</td>
<td>No failover</td>
<td>Mark failover interface as failed</td>
<td>Mark failover interface as failed</td>
<td>You should restore the failover link as soon as possible because the unit cannot fail over to the standby unit while the failover link is down.</td>
</tr>
<tr>
<td>Failover link failed at startup</td>
<td>No failover</td>
<td>Mark failover interface as failed</td>
<td>Become active</td>
<td>If the failover link is down at startup, both units become active.</td>
</tr>
<tr>
<td>Stateful Failover link failed</td>
<td>No failover</td>
<td>No action</td>
<td>No action</td>
<td>State information becomes out of date, and sessions are terminated if a failover occurs.</td>
</tr>
<tr>
<td>Interface failure on active unit above threshold</td>
<td>Failover</td>
<td>Mark active as failed</td>
<td>Become active</td>
<td>None.</td>
</tr>
<tr>
<td>Interface failure on standby unit above threshold</td>
<td>No failover</td>
<td>No action</td>
<td>Mark standby as failed</td>
<td>When the standby unit is marked as failed, then the active unit does not attempt to fail over even if the interface failure threshold is surpassed.</td>
</tr>
</tbody>
</table>
Optional Active/Standby Failover Settings

You can configure the following Active/Standby failover options when you initially configuring failover or after failover has been configured:

- HTTP replication with Stateful Failover—Allows connections to be included in the state information replication.
- Interface monitoring—Allows you to monitor up to 250 interfaces on a unit and control which interfaces affect your failover.
- Interface health monitoring—Enables the ASASM to detect and respond to interface failures more quickly.
- Failover criteria setup—Allows you to specify a specific number of interfaces or a percentage of monitored interfaces that must fail before failover occurs.
- Virtual MAC address configuration—Ensures that the secondary unit uses the correct MAC addresses when it is the active unit, even if it comes online before the primary unit.

Licensing Requirements for Active/Standby Failover

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

Prerequisites for Active/Standby Failover

Active/Standby failover has the following prerequisites:

- Both units must be identical ASASMs that are connected to each other through a dedicated failover link and, optionally, a Stateful Failover link.
- Both units must have the same software configuration and the proper license.
- Both units must be in the same mode (single or multiple, transparent or routed).

Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

Context Mode Guidelines

- Supported in single and multiple context mode.
- For multiple context mode, perform all steps in the system execution space unless otherwise noted.

Firewall Mode Guidelines

- Supported in transparent and routed firewall mode.
Chapter 50      Configuring Active/Standby Failover

IPv6 Guidelines
- IPv6 failover is supported.

Model Guidelines
- Stateful failover is not supported on the ASA 5505.

Additional Guidelines and Limitations
Configuring port security on the switch(es) connected to an ASASM failover pair can cause communication problems when a failover event occurs. This is because if a secure MAC address configured or learned on one secure port moves to another secure port, a violation is flagged by the switch port security feature.

ASASM failover replication fails if you try to make a configuration change in two or more contexts at the same time. The workaround is to make configuration changes on each unit sequentially.

The following guidelines and limitations apply for Active/Standby failover:
- To receive packets from both units in a failover pair, standby IP addresses need to be configured on all interfaces.
- The standby IP addresses are used on the ASASM that is currently the standby unit, and they must be in the same subnet as the active IP address on the corresponding interface on the active unit.
- If you change the console terminal pager settings on the active unit in a failover pair, the active console terminal pager settings change, but the standby unit settings do not. A default configuration issued on the active unit does affect behavior on the standby unit.
- When you enable interface monitoring, you can monitor up to 250 interfaces on a unit.
- By default, the ASASM does not replicate HTTP session information when Stateful Failover is enabled. Because HTTP sessions are typically short-lived, and because HTTP clients typically retry failed connection attempts, not replicating HTTP sessions increases system performance without causing serious data or connection loss. The failover replication http command enables the stateful replication of HTTP sessions in a Stateful Failover environment, but it could have a negative impact upon system performance.
- AnyConnect images must be the same on both ASAs in a failover pair. If the failover pair has mismatched images when a hitless upgrade is performed, then the WebVPN connection terminates in the final reboot step of the upgrade process, the database shows an orphaned session, and the IP pool shows that the IP address assigned to the client is “in use.”

Configuring Active/Standby Failover

This section describes how to configure Active/Standby failover. This section includes the following topics:
- Task Flow for Configuring Active/Standby Failover, page 50-8
- Configuring the Primary Unit, page 50-8
- Configuring the Secondary Unit, page 50-11
- Configuring Optional Active/Standby Failover Settings, page 50-12
Task Flow for Configuring Active/Standby Failover

To configure Active/Standby failover, perform the following steps:

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>Configure the primary unit, as shown in the “Configuring the Primary Unit” section on page 50-8.</td>
</tr>
<tr>
<td>Step 2</td>
<td>Configure the secondary unit, as shown in the “Configuring the Secondary Unit” section on page 50-11.</td>
</tr>
<tr>
<td>Step 3</td>
<td>(Optional) Configure optional Active/Standby failover settings, as shown in the “Configuring Optional Active/Standby Failover Settings” section on page 50-12.</td>
</tr>
</tbody>
</table>

Configuring the Primary Unit

Follow the steps in this section to configure the primary unit in a LAN-based, Active/Standby failover configuration. These steps provide the minimum configuration needed to enable failover on the primary unit.

Restrictions

Do not configure an IP address in interface configuration mode for the Stateful Failover link if you are going to use a dedicated Stateful Failover interface. You use the `failover interface ip` command to configure a dedicated Stateful Failover interface in a later step.

Prerequisites

- Configure standby addresses for all IP addresses according to Chapter 7, “Configuring Interfaces (Routed Mode),” or Chapter 8, “Configuring Interfaces (Transparent Mode).”
- For multiple context mode, complete this procedure in the system execution space. To change from the context to the system execution space, enter the `changeto system` command.
### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1**
 failover lan unit primary                  | Designates the unit as the primary unit.                                |
| **Step 2**
 failover lan interface *if_name*            | Specifies the interface to be used as the failover interface. This     |
| interface_id                                 | interface should not be used for any other purpose (except,          |
|                                             | optionally, the Stateful Failover link).                              |
| Example:
 hostname(config)# failover lan interface| The *if_name* argument assigns a name to the interface specified by    |
| folink GigabitEthernet0/3                   | the *interface_id* argument.                                           |
|                                             | The interface ID can be a physical interface or a redundant          |
|                                             | interface. On the ASA 5505 or ASASM, the *interface_id* specifies    |
|                                             | a VLAN.                                                                |
| **Note**                                     | Although you can use an EtherChannel as a failover or state link, to   |
|                                             | prevent out-of-order packets, only one interface in the EtherChannel  |
|                                             | is used. If that interface fails, then the next interface in the      |
|                                             | EtherChannel is used. You cannot alter the EtherChannel configuration |
|                                             | while it is in use as a failover link. To alter the configuration,    |
|                                             | you need to either shut down the EtherChannel while you make         |
|                                             | changes, or temporarily disable failover; either action prevents     |
|                                             | failover from occurring for the duration.                            |
| **Step 3**
 failover interface ip *if_name* [ip_address |
| mask standby ip_address | Assigns the active and standby IP addresses to the failover link.      |
| ipv6_address/prefix standby ipv6_address]    | You can assign either an IPv4 or an IPv6 address to the interface.     |
| Example:
 hostname(config)# failover interface ip | You cannot assign both types of addresses to the failover link.       |
| folink 172.27.48.1 255.255.255.0 standby     | The standby IP address must be in the same subnet as the active        |
| 172.27.48.2                                  | IP address. You do not need to identify the standby address subnet    |
| hostname(config)# failover interface ip      | mask.                                                                  |
| folink 2001:a0a:b00::a0a:b70/64 standby      | The failover link IP address and MAC address do not change at         |
| 2001:a0a:b00::a0a:b71                        | failover. The active IP address for the failover link always stays    |
|                                             | with the primary unit, while the standby IP address stays with the    |
|                                             | secondary unit.                                                      |
| **Step 4**
 interface *interface_id*                 | Enables the interface.                                                |
| no shutdown                                 |                                                                         |
| Example:
 hostname(config)# interface vlan100     |                                                                         |
| hostname(config-if)# no shutdown             |                                                                         |
## Configuring Active/Standby Failover

### Step 5

**Command**

```
failover link if_name interface_id
```

**Example:**

```
hostname(config)# failover link statelink GigabitEthernet0/2
```

(Optional) Specifies the interface to be used as the Stateful Failover link. This interface should not be used for any other purpose (except, optionally, the failover link).

**Note**

- If the Stateful Failover link uses the failover link or a data interface, then you only need to supply the `if_name` argument.
- The `if_name` argument assigns a logical name to the interface specified by the `interface_id` argument. The `interface_id` argument can be the physical port name, such as Ethernet1, or a previously created subinterface, such as Ethernet0/2.3. This interface can be a physical interface or a redundant interface.
- Although you can use an EtherChannel as a failover or state link, to prevent out-of-order packets, only one interface in the EtherChannel is used. If that interface fails, then the next interface in the EtherChannel is used.
- You cannot alter the EtherChannel configuration while it is in use as a failover link. To alter the configuration, you need to either shut down the EtherChannel while you make changes, or temporarily disable failover; either action prevents failover from occurring for the duration.

### Step 6

**Command**

```
failover interface ip if_name [ip_address mask standby ip_address | ipv6_address/prefix standby ipv6_address]
```

**Example:**

```
hostname(config)# failover interface ip folink 172.27.48.1 255.255.255.0 standby 172.27.48.2
hostname(config)# failover interface ip statelink 2001:a1a:b00::a0a:a70/64 standby 2001:a1a:b00::a0a:a71
```

(Optional) Assigns an active and standby IP address to the Stateful Failover link. You can assign either an IPv4 or an IPv6 address to the interface. You cannot assign both types of addresses to the Stateful Failover link.

**Note**

- If the stateful Failover link uses the failover link or data interface, skip this step. You have already defined the active and standby IP addresses for the interface.
- The standby IP address must be in the same subnet as the active IP address. You do not need to identify the standby address subnet mask.
- The Stateful Failover link IP address and MAC address do not change at failover unless it uses a data interface. The active IP address always stays with the primary unit, while the standby IP address stays with the secondary unit.

### Step 7

**Command**

```
interface interface_id no shutdown
```

**Example:**

```
hostname(config)# interface vlan100
hostname(config-if)# no shutdown
```

(Optional) Enables the interface.

If the Stateful Failover link uses the failover link or a data interface, skip this step. You have already enabled the interface.
Configuring Active/Standby Failover

Chapter 50      Configuring Active/Standby Failover

Configuring the Secondary Unit

The only configuration required on the secondary unit is for the failover interface. The secondary unit requires these commands to communicate initially with the primary unit. After the primary unit sends its configuration to the secondary unit, the only permanent difference between the two configurations is the `failover lan unit` command, which identifies each unit as primary or secondary.

Prerequisites

When configuring LAN-based failover, you must bootstrap the secondary device to recognize the failover link before the secondary device can obtain the running configuration from the primary device.

Detailed Steps

To configure the secondary unit, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 8</strong></td>
<td></td>
</tr>
<tr>
<td>failover</td>
<td>Enables failover.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# failover</td>
<td></td>
</tr>
<tr>
<td><strong>Step 9</strong></td>
<td></td>
</tr>
<tr>
<td>copy running-config startup-config</td>
<td>Saves the system configuration to flash memory.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# copy running-config startup-config</td>
<td></td>
</tr>
</tbody>
</table>

### Command Purpose

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td>failover lan interface <code>if_name</code> <code>interface_id</code></td>
<td>Specifies the interface to be used as the failover interface. (Use the same settings that you used for the primary unit.) The <code>if_name</code> argument assigns a name to the interface specified by the <code>interface_id</code> argument. The interface ID can be a physical interface or a redundant interface. EtherChannel interfaces are not supported.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# failover lan interface folink vlan100</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td>failover interface ip <code>if_name</code> `[ip_address mask standby ip_address</td>
<td>ipv6_address/prefix standby ipv6_address]`</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# failover interface ip folink 172.27.48.1 255.255.255.0 standby 172.27.48.2</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# failover interface ip folink 2001:a0a:b00::a0a:b70/64 standby 2001:a0a:b00::a0a:b71</td>
<td></td>
</tr>
<tr>
<td><strong>Note</strong></td>
<td>Enter this command exactly as you entered it on the primary unit when you configured the failover interface on the primary unit (including the same IP address).</td>
</tr>
</tbody>
</table>
### Configuring Active/Standby Failover

#### Step 3
```conf
interface interface_id
no shutdown
```

- **Purpose**: Enables the interface.

- **Example**:
  ```bash
  hostname(config)# interface vlan100
  hostname(config-if)# no shutdown
  ```

#### Step 4
```conf
failover lan unit secondary
```

- **Purpose**: (Optional) Designates this unit as the secondary unit.

- **Note**: This step is optional because, by default, units are designated as secondary unless previously configured.

#### Step 5
```conf
failover
```

- **Purpose**: Enables failover.

- **Example**:
  ```bash
  hostname(config)# failover
  ```

#### Step 6
```conf
copy running-config startup-config
```

- **Purpose**: Saves the configuration to flash memory.

- **Example**: 
  ```bash
  hostname(config)# copy running-config startup-config
  ```

### Configuring Optional Active/Standby Failover Settings

This section includes the following topics:

- Enabling HTTP Replication with Stateful Failover, page 50-13
- Disabling and Enabling Interface Monitoring, page 50-13
- Configuring Failover Criteria, page 50-14
- Configuring the Unit and Interface Health Poll Times, page 50-14
- Configuring Virtual MAC Addresses, page 50-15

You can configure the optional Active/Standby failover settings when initially configuring the primary unit in a failover pair (see Configuring the Primary Unit, page 50-8) or on the active unit in the failover pair after the initial configuration.
Enabling HTTP Replication with Stateful Failover

To allow HTTP connections to be included in the state information replication, you need to enable HTTP replication. Because HTTP connections are typically short-lived, and because HTTP clients typically retry failed connection attempts, HTTP connections are not automatically included in the replicated state information.

To enable HTTP state replication when Stateful Failover is enabled, enter the following command in global configuration mode:

```
failover replication http
```

Example:
```
hostname (config)# failover replication http
```

Disabling and Enabling Interface Monitoring

You can control which interfaces affect your failover policy by disabling the monitoring of specific interfaces and enabling the monitoring of others. This feature enables you to exclude interfaces attached to less critical networks from affecting your failover policy.

You can monitor up to 250 interfaces on a unit. By default, monitoring physical interfaces is enabled and monitoring subinterfaces is disabled.

Hello messages are exchanged during every interface poll frequency time period between the ASASM failover pair. The failover interface poll time is 3 to 15 seconds. For example, if the poll time is set to 5 seconds, testing begins on an interface if 5 consecutive hellos are not heard on that interface (25 seconds).

Monitored failover interfaces can have the following status:

- **Unknown**—Initial status. This status can also mean the status cannot be determined.
- **Normal**—The interface is receiving traffic.
- **Testing**—Hello messages are not heard on the interface for five poll times.
- **Link Down**—The interface or VLAN is administratively down.
- **No Link**—The physical link for the interface is down.
- **Failed**—No traffic is received on the interface, yet traffic is heard on the peer interface.

To enable or disable health monitoring for specific interfaces on units in single configuration mode, enter one of the following commands. Alternately, for units in multiple configuration mode, you must enter the commands within each security context.

Do one of the following:
Configuring Active/Standby Failover

Configuring Failover Criteria

You can specify a specific number of interface or a percentage of monitored interfaces that must fail before failover occurs. By default, a single interface failure causes failover.

To change the default failover criteria, enter the following command in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>failover interface-policy num[%]</td>
<td>Changes the default failover criteria.</td>
</tr>
<tr>
<td></td>
<td>When specifying a specific number of interfaces, the num argument can be from 1 to 250.</td>
</tr>
<tr>
<td></td>
<td>When specifying a percentage of interfaces, the num argument can be from 1 to 100.</td>
</tr>
</tbody>
</table>

Configuring the Unit and Interface Health Poll Times

The ASASM sends hello packets out of each data interface to monitor interface health. The appliance sends hello messages across the failover link to monitor unit health. If the ASASM does not receive a hello packet from the corresponding interface on the peer unit for over half of the hold time, then the additional interface testing begins. If a hello packet or a successful test result is not received within the specified hold time, the interface is marked as failed. Failover occurs if the number of failed interfaces meets the failover criteria.

Decreasing the poll and hold times enables the ASASM to detect and respond to interface failures more quickly but may consume more system resources. Increasing the poll and hold times prevents the ASASM from failing over on networks with higher latency.

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>no monitor-interface if_name</td>
<td>Disables health monitoring for an interface.</td>
</tr>
<tr>
<td></td>
<td>Example: hostname(config)# no monitor-interface lanlink</td>
</tr>
<tr>
<td>monitor-interface if_name</td>
<td>Enables health monitoring for an interface.</td>
</tr>
<tr>
<td></td>
<td>Example: hostname(config)# monitor-interface lanlink</td>
</tr>
</tbody>
</table>
Configuring Virtual MAC Addresses

In Active/Standby failover, the MAC addresses for the primary unit are always associated with the active IP addresses. If the secondary unit boots first and becomes active, it uses the burned-in MAC address for its interfaces. When the primary unit comes online, the secondary unit obtains the MAC addresses from the primary unit. The change can disrupt network traffic.

You can configure virtual MAC addresses for each interface to ensure that the secondary unit uses the correct MAC addresses when it is the active unit, even if it comes online before the primary unit. If you do not specify virtual MAC addresses the failover pair uses the burned-in NIC addresses as the MAC addresses.

Note

You cannot configure a virtual MAC address for the failover or Stateful Failover links. The MAC and IP addresses for those links do not change during failover.

To configure the virtual MAC addresses for an interface, enter the following command on the active unit:
Controlling Failover

This sections describes how to control and monitor failover. This section includes the following topics:

- Forcing Failover, page 50-16
- Disabling Failover, page 50-17
- Restoring a Failed Unit, page 50-17

Forcing Failover

To force the standby unit to become active, enter one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>failover active</td>
<td>Forces a failover when entered on the standby unit in a failover pair. The standby unit becomes the active unit.</td>
</tr>
</tbody>
</table>

Example:

```
hostname# failover active
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>no failover active</td>
<td>Forces a failover when entered on the active unit in a failover pair. The active unit becomes the standby unit.</td>
</tr>
</tbody>
</table>

Example:

```
hostname# no failover active
```
Disabling Failover

To disable failover, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>no failover</td>
<td>Disables failover. Disabling failover on an Active/Standby pair causes the active and standby state of each unit to be maintained until you restart. For example, the standby unit remains in standby mode so that both units do not start passing traffic. To make the standby unit active (even with failover disabled), see the &quot;Forcing Failover&quot; section on page 50-16.</td>
</tr>
</tbody>
</table>

Example:

```
hostname(config)# no failover
```

Restoring a Failed Unit

To restore a failed unit to an unfailed state, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>failover reset</td>
<td>Restores a failed unit to an unfailed state. Restoring a failed unit to an unfailed state does not automatically make it active; restored units remain in the standby state until made active by failover (forced or natural).</td>
</tr>
</tbody>
</table>

Example:

```
hostname(config)# failover reset
```

Testing the Failover Functionality

To test failover functionality, perform the following steps:

Step 1  Test that your active unit is passing traffic as expected by using FTP (for example) to send a file between hosts on different interfaces.

Step 2  Force a failover by entering the following command on the active unit:

```
hostname(config)# no failover active
```

Step 3  Use FTP to send another file between the same two hosts.

Step 4  If the test was not successful, enter the `show failover` command to check the failover status.

Step 5  When you are finished, you can restore the unit to active status by enter the following command on the newly active unit:

```
hostname(config)# no failover active
```

Note

When an ASASM interface goes down, for failover it is still considered to be a unit issue. If the ASASM detects that an interface is down, failover occurs immediately, without waiting for the interface holdtime. The interface holdtime is only useful when the ASASM considers its status to be OK, although it is not receiving hello packets from the peer. To simulate interface holdtime, shut down the VLAN on the switch to prevent peers from receiving hello packets from each other.
Monitoring Active/Standby Failover

After a failover event you should either re-launch ASDM or switch to another device in the Devices pane and then come back to the original ASA to continue monitoring the device. This action is necessary because the monitoring connection does not become re-established when ASDM is disconnected from and then reconnected to the device.

To monitor Active/Standby failover, enter one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show failover</code></td>
<td>Displays information about the failover state of the unit.</td>
</tr>
<tr>
<td><code>show monitor-interface</code></td>
<td>Displays information about the monitored interface.</td>
</tr>
<tr>
<td><code>show running-config failover</code></td>
<td>Displays the failover commands in the running configuration.</td>
</tr>
</tbody>
</table>

For more information about the output of the monitoring commands, refer to the Cisco ASA 5500 Series Command Reference.

Feature History for Active/Standby Failover

Table 50-3 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>This feature was introduced.</td>
<td>7.0</td>
<td>This feature was introduced.</td>
</tr>
<tr>
<td>IPv6 support for failover added.</td>
<td>8.2(2)</td>
<td>We modified the following commands: <code>failover interface ip</code>, <code>show failover</code>, <code>ipv6 address</code>, <code>show monitor-interface</code>.</td>
</tr>
</tbody>
</table>
Configuring Active/Active Failover

This chapter describes how to configure Active/Active failover and includes the following sections:

- Information About Active/Active Failover, page 51-1
- Licensing Requirements for Active/Active Failover, page 51-6
- Prerequisites for Active/Active Failover, page 51-6
- Guidelines and Limitations, page 51-7
- Configuring Active/Active Failover, page 51-8
- Remote Command Execution, page 51-21
- Monitoring Active/Active Failover, page 51-25
- Feature History for Active/Active Failover, page 51-25

Information About Active/Active Failover

This section describes Active/Active failover. This section includes the following topics:

- Active/Active Failover Overview, page 51-1
- Primary/Secondary Status and Active/Standby Status, page 51-2
- Device Initialization and Configuration Synchronization, page 51-3
- Command Replication, page 51-3
- Failover Triggers, page 51-4
- Failover Actions, page 51-5

Active/Active Failover Overview

Active/Active failover is only available to ASASMs in multiple context mode. In an Active/Active failover configuration, both ASASMs can pass network traffic.

In Active/Active failover, you divide the security contexts on the ASASM into failover groups. A failover group is simply a logical group of one or more security contexts. You can create a maximum of two failover groups. The admin context is always a member of failover group 1. Any unassigned security contexts are also members of failover group 1 by default.
The failover group forms the base unit for failover in Active/Active failover. Interface failure monitoring, failover, and active/standby status are all attributes of a failover group rather than the unit. When an active failover group fails, it changes to the standby state while the standby failover group becomes active. The interfaces in the failover group that becomes active assume the MAC and IP addresses of the interfaces in the failover group that failed. The interfaces in the failover group that is now in the standby state take over the standby MAC and IP addresses.

**Note**

A failover group failing on a unit does not mean that the unit has failed. The unit may still have another failover group passing traffic on it.

When creating the failover groups, you should create them on the unit that will have failover group 1 in the active state.

**Note**

Active/Active failover generates virtual MAC addresses for the interfaces in each failover group. If you have more than one Active/Active failover pair on the same network, it is possible to have the same default virtual MAC addresses assigned to the interfaces on one pair as are assigned to the interfaces of the other pairs because of the way the default virtual MAC addresses are determined. To avoid having duplicate MAC addresses on your network, make sure you assign each physical interface a virtual active and standby MAC address.

### Primary/Secondary Status and Active/Standby Status

As in Active/Standby failover, one unit in an Active/Active failover pair is designated the primary unit, and the other unit the secondary unit. Unlike Active/Standby failover, this designation does not indicate which unit becomes active when both units start simultaneously. Instead, the primary/secondary designation does two things:

- Determines which unit provides the running configuration to the pair when they boot simultaneously.
- Determines on which unit each failover group appears in the active state when the units boot simultaneously. Each failover group in the configuration is configured with a primary or secondary unit preference. You can configure both failover groups be in the active state on a single unit in the pair, with the other unit containing the failover groups in the standby state. However, a more typical configuration is to assign each failover group a different role preference to make each one active on a different unit, distributing the traffic across the devices.

Which unit each failover group becomes active on is determined as follows:

- When a unit boots while the peer unit is not available, both failover groups become active on the unit.
- When a unit boots while the peer unit is active (with both failover groups in the active state), the failover groups remain in the active state on the active unit regardless of the primary or secondary preference of the failover group until one of the following occurs:
  - A failover occurs.
  - You manually force a failover.
  - You configured preemption for the failover group, which causes the failover group to automatically become active on the preferred unit when the unit becomes available.
- When both units boot at the same time, each failover group becomes active on its preferred unit after the configurations have been synchronized.
Device Initialization and Configuration Synchronization

Configuration synchronization occurs when one or both units in a failover pair boot. The configurations are synchronized as follows:

- When a unit boots while the peer unit is active (with both failover groups active on it), the booting unit contacts the active unit to obtain the running configuration regardless of the primary or secondary designation of the booting unit.
- When both units boot simultaneously, the secondary unit obtains the running configuration from the primary unit.

When the replication starts, the ASASM console on the unit sending the configuration displays the message “Beginning configuration replication: Sending to mate,” and when it is complete, the ASASM displays the message “End Configuration Replication to mate.” During replication, commands entered on the unit sending the configuration may not replicate properly to the peer unit, and commands entered on the unit receiving the configuration may be overwritten by the configuration being received. Avoid entering commands on either unit in the failover pair during the configuration replication process. Depending upon the size of the configuration, replication can take from a few seconds to several minutes.

On the unit receiving the configuration, the configuration exists only in running memory. To save the configuration to flash memory after synchronization enter the `write memory all` command in the system execution space on the unit that has failover group 1 in the active state. The command is replicated to the peer unit, which proceeds to write its configuration to flash memory. Using the `all` keyword with this command causes the system and all context configurations to be saved.

**Note**

Startup configurations saved on external servers are accessible from either unit over the network and do not need to be saved separately for each unit. Alternatively, you can copy the contexts configuration files from the disk on the primary unit to an external server, and then copy them to disk on the secondary unit, where they become available when the unit reloads.

Command Replication

After both units are running, commands are replicated from one unit to the other as follows:

- Commands entered within a security context are replicated from the unit on which the security context appears in the active state to the peer unit.

**Note**

A context is considered in the active state on a unit if the failover group to which it belongs is in the active state on that unit.

- Commands entered in the system execution space are replicated from the unit on which failover group 1 is in the active state to the unit on which failover group 1 is in the standby state.
- Commands entered in the admin context are replicated from the unit on which failover group 1 is in the active state to the unit on which failover group 1 is in the standby state.

Failure to enter the commands on the appropriate unit for command replication to occur causes the configurations to be out of synchronization. Those changes may be lost the next time the initial configuration synchronization occurs.

**Table 51-1** lists the commands that are and are not replicated to the standby unit.
Information About Active/Active Failover

You can use the write standby command to resynchronize configurations that have become out of sync. For Active/Active failover, the write standby command behaves as follows:

- If you enter the write standby command in the system execution space, the system configuration and the configurations for all security contexts on the ASASM are written to the peer unit. This includes configuration information for security contexts that are in the standby state. You must enter the command in the system execution space on the unit that has failover group 1 in the active state.

  Note. If there are security contexts in the active state on the peer unit, the write standby command causes active connections through those contexts to be terminated. Use the failover active command on the unit providing the configuration to make sure all contexts are active on that unit before entering the write standby command.

- If you enter the write standby command in a security context, only the configuration for the security context is written to the peer unit. You must enter the command in the security context on the unit where the security context appears in the active state.

Replicated commands are not saved to the flash memory when replicated to the peer unit. They are added to the running configuration. To save replicated commands to flash memory on both units, use the write memory or copy running-config startup-config command on the unit that you made the changes on. The command is replicated to the peer unit and cause the configuration to be saved to flash memory on the peer unit.

### Failover Triggers

In Active/Active failover, failover can be triggered at the unit level if one of the following events occurs:

- The unit has a hardware failure.
- The unit has a power failure.
- The unit has a software failure.
- You force a failover. (See Forcing Failover, page 51-24.)

Failover is triggered at the failover group level when one of the following events occurs:

- Too many monitored interfaces in the group fail.

---

**Table 51-1 Command Replication**

<table>
<thead>
<tr>
<th>Commands Replicated to the Standby Unit</th>
<th>Commands Not Replicated to the Standby Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>All configuration commands except for mode, firewall, and failover lan unit</td>
<td>All forms of the copy command except for copy running-config startup-config</td>
</tr>
<tr>
<td>copy running-config startup-config</td>
<td>All forms of the write command except for write memory</td>
</tr>
<tr>
<td>delete</td>
<td>debug</td>
</tr>
<tr>
<td>mkdir</td>
<td>failover lan unit</td>
</tr>
<tr>
<td>rename</td>
<td>firewall</td>
</tr>
<tr>
<td>rmkdir</td>
<td>mode</td>
</tr>
<tr>
<td>write memory</td>
<td>show</td>
</tr>
</tbody>
</table>

---

Note. If there are security contexts in the active state on the peer unit, the write standby command causes active connections through those contexts to be terminated. Use the failover active command on the unit providing the configuration to make sure all contexts are active on that unit before entering the write standby command.
You force a failover. (See Forcing Failover, page 51-24.)

You configure the failover threshold for each failover group by specifying the number or percentage of interfaces within the failover group that must fail before the group fails. Because a failover group can contain multiple contexts, and each context can contain multiple interfaces, it is possible for all interfaces in a single context to fail without causing the associated failover group to fail.

See the “Failover Health Monitoring” section on page 49-16 for more information about interface and unit monitoring.

**Failover Actions**

In an Active/Active failover configuration, failover occurs on a failover group basis, not a system basis. For example, if you designate both failover groups as active on the primary unit, and failover group 1 fails, then failover group 2 remains active on the primary unit while failover group 1 becomes active on the secondary unit.

**Note**

When configuring Active/Active failover, make sure that the combined traffic for both units is within the capacity of each unit.

Table 51-2 shows the failover action for each failure event. For each failure event, the policy (whether or not failover occurs), actions for the active failover group, and actions for the standby failover group are given.

<table>
<thead>
<tr>
<th>Failure Event</th>
<th>Policy</th>
<th>Active Group Action</th>
<th>Standby Group Action</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>A unit experiences a power or software failure</td>
<td>Failover</td>
<td>Become standby Mark as failed</td>
<td>Become active Mark as failed</td>
<td>When a unit in a failover pair fails, any active failover groups on that unit are marked as failed and become active on the peer unit.</td>
</tr>
<tr>
<td>Interface failure on active failover group above threshold</td>
<td>Failover</td>
<td>Mark active group as failed</td>
<td>Become active</td>
<td>None.</td>
</tr>
<tr>
<td>Interface failure on standby failover group above threshold</td>
<td>No failover</td>
<td>No action</td>
<td>Mark standby group as failed</td>
<td>When the standby failover group is marked as failed, the active failover group does not attempt to fail over, even if the interface failure threshold is surpassed.</td>
</tr>
<tr>
<td>Formerly active failover group recovers</td>
<td>No failover</td>
<td>No action</td>
<td>No action</td>
<td>Unless failover group preemption is configured, the failover groups remain active on their current unit.</td>
</tr>
<tr>
<td>Failover link failed at startup</td>
<td>No failover</td>
<td>Become active</td>
<td>Become active</td>
<td>If the failover link is down at startup, both failover groups on both units become active.</td>
</tr>
</tbody>
</table>
Optional Active/Active Failover Settings

You can configure the following Active/Standby failover options when you initially configuring failover or after failover has been configured:

- **Failover Group Preemption**—Assigns a primary or secondary priority to a failover group to specify on which unit in the failover group becomes active when both units boot simultaneously.

- **HTTP replication with Stateful Failover**—Allows connections to be included in the state information replication.

- **Interface monitoring**—Allows you to monitor up to 250 interfaces on a unit and control which interfaces affect your failover.

- **Interface health monitoring**—Enables the security appliance to detect and respond to interface failures more quickly.

- **Failover criteria setup**—Allows you to specify a specific number of interfaces or a percentage of monitored interfaces that must fail before failover occurs.

- **Virtual MAC address configuration**—Ensures that the secondary unit uses the correct MAC addresses when it is the active unit, even if it comes online before the primary unit.

**Licensing Requirements for Active/Active Failover**

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

**Prerequisites for Active/Active Failover**

In Active/Active failover, both units must have the following:

- The same hardware model.

- The same number of interfaces.
The same types of interfaces.
The same software version, with the same major (first number) and minor (second number) version numbers. However you can use different versions of the software during an upgrade process; for example you can upgrade one unit from Version 7.0(1) to Version 7.9(2) and have failover remain active. We recommend upgrading both units to the same version to ensure long-term compatibility.
The same software configuration.
The same mode (multiple context mode).
The proper license.

Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

Context Mode Guidelines
Supported in multiple context mode only.

Firewall Mode Guidelines
Supported in routed and transparent firewall mode.

IPv6 Guidelines
IPv6 failover is supported.

Model Guidelines
Active/Active failover is not available on the Cisco ASA 5505.

Additional Guidelines and Limitations
No two interfaces in the same context should be configured in the same ASR group.
ASASM failover replication fails if you try to make a configuration change on two or more contexts at the same time. The workaround is to make configuration changes on each unit sequentially.
The following features are not supported for Active/Active failover:
- To receive packets from both units in a failover pair, standby IP addresses need to be configured on all interfaces.
- The standby IP address is used on the security appliance that is currently the standby unit, and it must be in the same subnet as the active IP address.
- You can define a maximum number of two failover groups.
- Failover groups can only be added to the system context of devices that are configured for multiple context mode.
- You can create and remove failover groups only when failover is disabled.
- Entering the failover group command puts you in the failover group command mode. The primary, secondary, preempt, replication http, interface-policy, mac address, and polltime interface commands are available in the failover group configuration mode. Use the exit command to return to global configuration mode.
The `failover polltime interface`, `failover interface-policy`, `failover replication http`, and `failover mac address` commands have no affect on Active/Active failover configurations. They are overridden by the following failover group configuration mode commands: `polltime interface`, `interface-policy`, `replication http`, and `mac address`.

When removing failover groups, you must remove failover group 1 last. Failover group 1 always contains the admin context. Any context not assigned to a failover group defaults to failover group 1. You cannot remove a failover group that has contexts explicitly assigned to it.

VPN failover is unavailable. (It is available in Active/Standby failover configurations only.)

## Configuring Active/Active Failover

This section describes how to configure Active/Active failover using an Ethernet failover link. When configuring LAN-based failover, you must bootstrap the secondary device to recognize the failover link before the secondary device can obtain the running configuration from the primary device.

This section includes the following topics:

- Task Flow for Configuring Active/Active Failover, page 51-8
- Configuring the Primary Failover Unit, page 51-8
- Configuring the Secondary Failover Unit, page 51-11

### Task Flow for Configuring Active/Active Failover

To configure Active/Active Failover, perform the following steps:

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>Configure the primary unit, as shown in the “Configuring the Primary Failover Unit” section on page 51-8.</td>
</tr>
<tr>
<td>Step 2</td>
<td>Configure the secondary unit, as shown in the “Configuring the Secondary Failover Unit” section on page 51-11.</td>
</tr>
<tr>
<td>Step 3</td>
<td>(Optional) Configure optional Active/Active failover settings, as shown in the “Optional Active/Active Failover Settings” section on page 51-6.</td>
</tr>
</tbody>
</table>

## Configuring the Primary Failover Unit

Follow the steps in this section to configure the primary unit in an Active/Active failover configuration. These steps provide the minimum configuration needed to enable failover on the primary unit.

### Restrictions

Do not configure an IP address for the Stateful Failover link if you are going to use a dedicated Stateful Failover interface. You use the `failover interface ip` command to configure a dedicated Stateful Failover interface in a later step.
Prerequisites

- Configure standby addresses for all IP addresses according to Chapter 7, “Configuring Interfaces (Routed Mode),” or Chapter 8, “Configuring Interfaces (Transparent Mode).”
- Complete this procedure in the system execution space. To change from the context to the system execution space, enter the `changeto system` command.

Detailed Steps

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><code>failover lan unit primary</code></td>
<td>Designates the unit as the primary unit.</td>
</tr>
<tr>
<td>2</td>
<td><code>failover lan interface if_name phy_if</code></td>
<td>Specifies the interface to be used as the failover interface. The <code>if_name</code> argument assigns a name to the interface specified by the <code>phy_if</code> argument. The <code>phy_if</code> argument can be the physical port name, such as Ethernet1, or a previously created subinterface, such as Ethernet0/2.3. On the ASASM, the <code>phy_if</code> specifies a VLAN. This interface should not be used for any other purpose (except, optionally, the Stateful Failover link).</td>
</tr>
<tr>
<td>3</td>
<td>`failover interface ip if_name [ip_address mask</td>
<td>ip_addr</td>
</tr>
<tr>
<td>4</td>
<td><code>failover link if_name phy_if</code></td>
<td>(Optional) Specifies the interface to be used as the Stateful Failover link.</td>
</tr>
</tbody>
</table>

**Note**

If the Stateful Failover link uses the failover link or a data interface, then you only need to supply the `if_name` argument.

The `if_name` argument assigns a logical name to the interface specified by the `phy_if` argument. The `phy_if` argument can be the physical port name, such as Ethernet1, or a previously created subinterface, such as Ethernet0/2.3. On the ASASM, the `phy_if` specifies a VLAN. This interface should not be used for any other purpose (except, optionally, the failover link).
### Command

**Step 5**

```plaintext
failover interface ip if_name [ip_address mask standby ip_address | ipv6_address/prefix standby ipv6_address]
```

**Purpose**

(Optional) Assigns an active and standby IP address to the Stateful Failover link. You can assign either an IPv4 or an IPv6 address to the interface. You cannot assign both types of addresses to the Stateful Failover link.

**Example:**

```
hostname(config)# failover interface ip
folink 172.27.48.1 255.255.255.0 standby 172.27.48.2

hostname(config)# failover interface ip
statelink 2001:a1a:b00::a0a:a70/64 standby 2001:a1a:b00::a0a:a71
```

**Note**

If the Stateful Failover link uses the failover link or data interface, skip this step. You have already defined the active and standby IP addresses for the interface.

The standby IP address must be in the same subnet as the active IP address. You do not need to identify the standby address subnet mask.

The Stateful Failover link IP address and MAC address do not change at failover unless it uses a data interface. The active IP address always stays with the primary unit, while the standby IP address stays with the secondary unit.

**Step 6**

```plaintext
interface phy_if
no shutdown
```

**Purpose**

Enables the interface.

**Note**

If the Stateful Failover link uses the failover link or regular data interface, skip this step. You have already enabled the interface.

**Step 7**

```plaintext
failover group {1 | 2}
primary | secondary
```

**Purpose**

Configures the failover groups.

You can have only two failover groups. The `failover group` command creates the specified failover group if it does not exist and enters the failover group configuration mode.

For each failover group, specify whether the failover group has primary or secondary preference using the `primary` or `secondary` commands. You can assign the same preference to both failover groups. For traffic sharing configurations, you should assign each failover group a different unit preference.

The `exit` command restores global configuration mode.

The example assigns failover group 1 as the primary preference and failover group 2 as the secondary preference.

**Step 8**

```plaintext
context name
join-failover-group {1 | 2}
```

**Purpose**

Assigns each user context to a failover group (in context configuration mode).

Any unassigned contexts are automatically assigned to failover group 1. The admin context is always a member of failover group 1.
Configuring the Secondary Failover Unit

Follow the steps in this section to configure the secondary unit in an Active/Active failover configuration. These steps provide the minimum configuration needed to enable failover on the secondary unit.

Detailed Steps

To configure the secondary failover unit, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Specifies the interface to be used as the failover interface.</td>
</tr>
<tr>
<td><code>failover lan interface if_name phy_if</code></td>
<td>The <code>if_name</code> argument assigns a name to the interface specified by the <code>phy_if</code> argument.</td>
</tr>
<tr>
<td>The <code>phy_if</code> argument can be the physical port name, such as Ethernet1, or a previously created subinterface, such as Ethernet0/2.3. On the ASASM, the <code>phy_if</code> specifies a VLAN. This interface should not be used for any other purpose (except, optionally, the Stateful Failover link).</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td>Assigns the active and standby IP addresses to the failover link.</td>
</tr>
<tr>
<td><code>hostname(config)# failover lan interface folink GigabitEthernet0/3</code></td>
<td>You can assign either an IPv4 or an IPv6 address to the interface. You cannot assign both types of addresses to the failover link.</td>
</tr>
<tr>
<td>The standby IP address must be in the same subnet as the active IP address. You do not need to identify the standby address subnet mask.</td>
<td></td>
</tr>
<tr>
<td><code>hostname(config)# failover interface ip folink 172.27.48.1 255.255.255.0 standby 172.27.48.2</code></td>
<td>The failover link IP address and MAC address do not change at failover. The active IP address for the failover link always stays with the primary unit, while the standby IP address stays with the secondary unit.</td>
</tr>
<tr>
<td><code>hostname(config)# failover interface ip folink 2001:a0a:b00::a0a:b70/64 standby 2001:a0a:b00::a0a:b71</code></td>
<td>Enables the interface.</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Enables the interface.</td>
</tr>
<tr>
<td><code>interface phy_if</code></td>
<td>Example:</td>
</tr>
<tr>
<td><code>hostname(config-if)# interface GigabitEthernet0/3</code></td>
<td><code>hostname(config)# failover interface ip folink GigabitEthernet0/3</code></td>
</tr>
</tbody>
</table>
Configuring Active/Active Failover

Configuring Optional Active/Active Failover Settings

The following optional Active/Active failover settings can be configured when you are initially configuring failover or after you have already established failover. Unless otherwise noted, the commands should be entered on the unit that has failover group 1 in the active state.

This section includes the following topics:

- Configuring Failover Group Preemption, page 51-12
- Enabling HTTP Replication with Stateful Failover, page 51-14
- Disabling and Enabling Interface Monitoring, page 51-14
- Configuring Interface Health Monitoring, page 51-15
- Configuring Failover Criteria, page 51-16
- Configuring Virtual MAC Addresses, page 51-16
- Configuring Support for Asymmetrically Routed Packets, page 51-18

Configuring Failover Group Preemption

Assigning a primary or secondary priority to a failover group specifies which unit the failover group becomes active on when both units boot simultaneously. However, if one unit boots before the other, then both failover groups become active on that unit. When the other unit comes online, any failover groups that have the unit as a priority do not become active on that unit unless manually forced over, unless a
failover occurs, or unless the failover group is configured with the `preempt` command. The `preempt` command causes a failover group to become active on the designated unit automatically when that unit becomes available.

To configure preemption for the specified failover group, enter the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td>`failover group {1</td>
<td>2}`</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>hostname(config)# failover group 1</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td><code>preempt [delay]</code></td>
<td>Causes the failover group to become active on the designated unit.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>hostname(config-fover-group)# preempt 1200</code></td>
<td>Causes the failover group to become active on the designated unit.</td>
</tr>
<tr>
<td></td>
<td>You can enter an optional <em>delay</em> value, which specifies the number</td>
</tr>
<tr>
<td></td>
<td>of seconds the failover group remains active on the current unit</td>
</tr>
<tr>
<td></td>
<td>before automatically becoming active on the designated unit.</td>
</tr>
<tr>
<td></td>
<td>Valid values are from 1 to 1200.</td>
</tr>
<tr>
<td><strong>Note</strong></td>
<td></td>
</tr>
<tr>
<td>If Stateful Failover is enabled, the preemption is delayed until the connections are replicated from the unit on which the failover group is currently active.</td>
<td></td>
</tr>
</tbody>
</table>

**Example**

The following example configures failover group 1 with the primary unit as the higher priority and failover group 2 with the secondary unit as the higher priority. Both failover groups are configured with the `preempt` command with a wait time of 100 seconds, so the groups will automatically become active on their preferred unit 100 seconds after the units become available.

```
hostname(config)# failover group 1
hostname(config-fover-group)# primary
hostname(config-fover-group)# preempt 100
hostname(config-fover-group)# exit
hostname(config)# failover group 2
hostname(config-fover-group)# secondary
hostname(config-fover-group)# preempt 100
hostname(config-fover-group)# mac-address e1 0000.a000.a011 0000.a000.a012
hostname(config-fover-group)# exit
hostname(config)#
```
Enabling HTTP Replication with Stateful Failover

To allow HTTP connections to be included in the state information, you need to enable HTTP replication. Because HTTP connections are typically short-lived, and because HTTP clients typically retry failed connection attempts, HTTP connections are not automatically included in the replicated state information.

You can use the `replication http` command to cause a failover group to replicate HTTP state information when Stateful Failover is enabled.

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td>failover group (1</td>
<td>2)</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# failover group 1</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td>replication http</td>
<td>Enables HTTP state replication for the specified failover group. This command affects only the failover group in which it was configured. To enable HTTP state replication for both failover groups you must enter this command in each group. This command should be entered in the system execution space.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config-fover-group)# replication http</td>
<td></td>
</tr>
</tbody>
</table>

Example

The following example shows a possible configuration for a failover group:

```plaintext
hostname(config)# failover group 1
hostname(config-fover-group)# primary
hostname(config-fover-group)# preempt 100
hostname(config-fover-group)# replication http
hostname(config-fover-group)# exit
```

Disabling and Enabling Interface Monitoring

You can control which interfaces affect your failover policy by disabling the monitoring of specific interfaces and enabling the monitoring of others. This feature enables you to exclude interfaces attached to less critical networks from affecting your failover policy.

You can monitor up to 250 interfaces on a unit. By default, monitoring physical interfaces is enabled and monitoring subinterfaces is disabled.

Hello messages are exchanged during every interface poll frequency time period between the security appliance failover pair. The failover interface poll time is 3 to 15 seconds. For example, if the poll time is set to 5 seconds, testing begins on an interface if 5 consecutive hellos are not heard on that interface (25 seconds).

Monitored failover interfaces can have the following status:

- **Unknown**—Initial status. This status can also mean the status cannot be determined.
- **Normal**—The interface is receiving traffic.
- **Testing**—Hello messages are not heard on the interface for five poll times.
- **Link Down**—The interface or VLAN is administratively down.
- **No Link**—The physical link for the interface is down.
Failed—No traffic is received on the interface, yet traffic is heard on the peer interface.

In Active/Active failover, this command is only valid within a context.

To enable or disable interface monitoring for specific interfaces, enter one of the following commands:

Do one of the following:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>no monitor-interface if_name</td>
<td>Disables health monitoring for an interface.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname/context (config)# no monitor-interface 1</td>
<td></td>
</tr>
<tr>
<td>monitor-interface if_name</td>
<td>Enables health monitoring for an interface.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname/context (config)# monitor-interface 1</td>
<td></td>
</tr>
</tbody>
</table>

Example

The following example enables monitoring on an interface named “inside”:

hostname(config)# monitor-interface inside
hostname(config)#

Configuring Interface Health Monitoring

The ASASM sends hello packets out of each data interface to monitor interface health. If the ASASM does not receive a hello packet from the corresponding interface on the peer unit for over half of the hold time, then the additional interface testing begins. If a hello packet or a successful test result is not received within the specified hold time, the interface is marked as failed. Failover occurs if the number of failed interfaces meets the failover criteria.

Decreasing the poll and hold times enables the ASASM to detect and respond to interface failures more quickly but may consume more system resources.

To change the default interface poll time, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 failover group (1</td>
<td>Specifies the failover group.</td>
</tr>
<tr>
<td>2)</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# failover group 1</td>
<td></td>
</tr>
<tr>
<td>Step 2 polltime interface seconds</td>
<td>Specifies the data interface poll and hold times in the Active/Active failover configuration.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config-fover-group)# polltime interface seconds</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Valid values for the poll time are from 1 to 15 seconds or, if the optional msec keyword is used, from 500 to 999 milliseconds. The hold time determines how long it takes from the time a hello packet is missed to when the interface is marked as failed. Valid values for the hold time are from 5 to 75 seconds. You cannot enter a hold time that is less than 5 times the poll time.</td>
</tr>
</tbody>
</table>


Configuring Active/Active Failover

Example

The following partial example shows a possible configuration for a failover group. The interface poll time is set to 500 milliseconds and the hold time to 5 seconds for data interfaces in failover group 1.

```
hostname(config)# failover group 1
hostname(config-fover-group)# primary
hostname(config-fover-group)# preempt 100
hostname(config-fover-group)# polltime interface msec 500 holdtime 5
hostname(config-fover-group)# exit
hostname(config)#
```

Configuring Failover Criteria

By default, if a single interface fails, failover occurs. You can specify a specific number of interfaces or a percentage of monitored interfaces that must fail before a failover occurs. The failover criteria is specified on a failover group basis.

To change the default failover criteria for the specified failover group, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> failover group {1</td>
<td>2}</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# failover group 1</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> interface-policy num[%]</td>
<td>Specifies the policy for failover when monitoring detects an interface failure.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config-fover-group)# interface-policy 225</td>
<td></td>
</tr>
</tbody>
</table>

The following partial example shows a possible configuration for a failover group:

```
hostname(config)# failover group 1
hostname(config-fover-group)# primary
hostname(config-fover-group)# preempt 100
hostname(config-fover-group)# interface-policy 25%
hostname(config-fover-group)# exit
hostname(config)#
```

Configuring Virtual MAC Addresses

Active/Active failover uses virtual MAC addresses on all interfaces. If you do not specify the virtual MAC addresses, then they are computed as follows:

- Active unit default MAC address: 00a0.c9physical_port_number.failover_group_id01
- Standby unit default MAC address: 00a0.c9physical_port_number.failover_group_id02
If you have more than one Active/Active failover pair on the same network, it is possible to have the same default virtual MAC addresses assigned to the interfaces on one pair as are assigned to the interfaces of the other pairs because of the way the default virtual MAC addresses are determined. To avoid having duplicate MAC addresses on your network, make sure you assign each physical interface a virtual active and standby MAC address for all failover groups.

There are multiple ways to configure virtual MAC addresses on the ASASM. When more than one method has been used to configure virtual MAC addresses, the ASASM uses the following order of preference to determine which virtual MAC address is assigned to an interface:

1. The `mac-address` command (in interface configuration mode) address
2. The `mac-address auto` command generate address
3. The `failover mac address` command or `mac address` command (in failover group configuration mode) address (used in the following procedure)
4. The automatically generated failover MAC address

Use the `show interface` command to display the MAC address used by an interface.

To configure specific active and standby MAC addresses for an interface, perform the following steps.

### Detailed Steps

#### Command

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>`failover group {1</td>
<td>2}`</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>hostname(config)# failover group 1</code></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td><code>mac address phy_if active_mac standby_mac</code></td>
<td>Specifies the virtual MAC addresses for the active and standby units.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>hostname(config-fover-group)# mac address gilabitethernet1/0 0000.a000.a011 0000.a000.a012</code></td>
<td>The <code>phy_if</code> argument is the physical name of the interface, such as GigabitEthernet1/0. On the ASASM, the <code>phy_if</code> specifies a VLAN. The <code>active_mac</code> and <code>standby_mac</code> arguments are MAC addresses in H.H.H format, where H is a 16-bit hexadecimal digit. For example, the MAC address 00-0C-F1-42-4C-DE would be entered as 000C.F142.4CDE. The <code>active_mac</code> address is associated with the active IP address for the interface, and the <code>standby_mac</code> is associated with the standby IP address for the interface.</td>
</tr>
</tbody>
</table>

#### Example

The following partial example shows a possible configuration for a failover group:

```
hostname(config)# failover group 1
hostname(config-fover-group)# primary
hostname(config-fover-group)# preempt 100
hostname(config-fover-group)# exit
hostname(config)# failover group 2
hostname(config-fover-group)# secondary
hostname(config-fover-group)# preempt 100
```
Configuring Support for Asymmetrically Routed Packets

When running in Active/Active failover, a unit may receive a return packet for a connection that originated through its peer unit. Because the ASASM that receives the packet does not have any connection information for the packet, the packet is dropped. This most commonly occurs when the two ASASMs in an Active/Active failover pair are connected to different service providers and the outbound connection does not use a NAT address.

You can prevent the return packets from being dropped using the `asr-group` command on interfaces where this is likely to occur. When an interface configured with the `asr-group` command receives a packet for which it has no session information, it checks the session information for the other interfaces that are in the same group. If it does not find a match, the packet is dropped. If it finds a match, then one of the following actions occurs:

- If the incoming traffic originated on a peer unit, some or all of the layer 2 header is rewritten and the packet is redirected to the other unit. This redirection continues as long as the session is active.
- If the incoming traffic originated on a different interface on the same unit, some or all of the layer 2 header is rewritten and the packet is reinjected into the stream.

**Note**

Using the `asr-group` command to configure asymmetric routing support is more secure than using the `static` command with the `nailed` option.

The `asr-group` command does not provide asymmetric routing; it restores asymmetrically routed packets to the correct interface.

**Prerequisites**

You must have the following configured for asymmetric routing support to function properly:

- Active/Active Failover
- Stateful Failover—Passes state information for sessions on interfaces in the active failover group to the standby failover group.
- Replication HTTP—HTTP session state information is not passed to the standby failover group, and therefore is not present on the standby interface. For the ASASM to be able to re-route asymmetrically routed HTTP packets, you need to replicate the HTTP state information.

You can configure the `asr-group` command on an interface without having failover configured, but it does not have any effect until Stateful Failover is enabled.

**Detailed Steps**

To configure support for asymmetrically routed packets, perform the following steps:

1. **Step 1** Configure Active/Active Stateful Failover for the failover pair. See the “Configuring Active/Active Failover” section on page 51-8.
Step 2  For each interface that you want to participate in asymmetric routing support, enter the following command. You must enter the command on the unit where the context is in the active state so that the command is replicated to the standby failover group. For more information about command replication, see Command Replication, page 51-3.

```
hostname/ctx(config)# interface phy_if
hostname/ctx(config-if)# asr-group num
```

Valid values for `num` range from 1 to 32. You need to enter the command for each interface that participates in the asymmetric routing group. You can view the number of ASR packets transmitted, received, or dropped by an interface using the `show interface detail` command. You can have more than one ASR group configured on the ASASM, but only one per interface. Only members of the same ASR group are checked for session information.

Example

**Figure 51-1** shows an example of using the `asr-group` command for asymmetric routing support.

![Figure 51-1 ASR Example](image)

The two units have the following configuration (configurations show only the relevant commands). The device labeled SecAppA in the diagram is the primary unit in the failover pair.

**Example 51-1  Primary Unit System Configuration**

```
hostname primary
interface GigabitEthernet0/1
description LAN/STATE Failover Interface
interface GigabitEthernet0/2
```

```
no shutdown
interface GigabitEthernet0/3
no shutdown
interface GigabitEthernet0/4
no shutdown
interface GigabitEthernet0/5
no shutdown
failover
failover lan unit primary
failover lan interface folink GigabitEthernet0/1
failover link folink
failover interface ip folink 10.0.4.1 255.255.255.0 standby 10.0.4.11
failover group 1
primary
failover group 2
secondary
admin-context admin
context admin
description admin
allocate-interface GigabitEthernet0/2
allocate-interface GigabitEthernet0/3
config-url flash:/admin.cfg
join-failover-group 1
context ctx1
description ctx1
allocate-interface GigabitEthernet0/4
allocate-interface GigabitEthernet0/5
config-url flash:/ctx1.cfg
join-failover-group 2

Example 51-2  admin Context Configuration

hostname SecAppA
interface GigabitEthernet0/2
nameif outsideISP-A
security-level 0
ip address 192.168.1.1 255.255.255.0 standby 192.168.1.2
asr-group 1
interface GigabitEthernet0/3
nameif inside
security-level 100
ip address 10.1.0.1 255.255.255.0 standby 10.1.0.11
monitor-interface outside

Example 51-3  ctx1 Context Configuration

hostname SecAppB
interface GigabitEthernet0/4
nameif outsideISP-B
security-level 0
ip address 192.168.2.2 255.255.255.0 standby 192.168.2.1
asr-group 1
interface GigabitEthernet0/5
nameif inside
security-level 100
ip address 10.2.20.1 255.255.255.0 standby 10.2.20.11
Figure 51-1 shows the ASR support working as follows:

1. An outbound session passes through ASASM SecAppA. It exits interface outsideISP-A (192.168.1.1).
2. Because of asymmetric routing configured somewhere upstream, the return traffic comes back through the interface outsideISP-B (192.168.2.2) on ASASM SecAppB.
3. Normally the return traffic would be dropped because there is no session information for the traffic on interface 192.168.2.2. However, the interface is configured with the command `asr-group 1`. The unit looks for the session on any other interface configured with the same ASR group ID.
4. The session information is found on interface outsideISP-A (192.168.1.2), which is in the standby state on the unit SecAppB. Stateful Failover replicated the session information from SecAppA to SecAppB.
5. Instead of being dropped, the layer 2 header is rewritten with information for interface 192.168.1.1 and the traffic is redirected out of the interface 192.168.1.2, where it can then return through the interface on the unit from which it originated (192.168.1.1 on SecAppA). This forwarding continues as needed until the session ends.

Remote Command Execution

Remote command execution lets you send commands entered at the command line to a specific failover peer.

Because configuration commands are replicated from the active unit or context to the standby unit or context, you can use the `failover exec` command to enter configuration commands on the correct unit, no matter which unit you are logged in to. For example, if you are logged in to the standby unit, you can use the `failover exec active` command to send configuration changes to the active unit. Those changes are then replicated to the standby unit. Do not use the `failover exec` command to send configuration commands to the standby unit or context; those configuration changes are not replicated to the active unit and the two configurations will no longer be synchronized.

Output from configuration, exec, and `show` commands is displayed in the current terminal session, so you can use the `failover exec` command to issue `show` commands on a peer unit and view the results in the current terminal.

You must have sufficient privileges to execute a command on the local unit to execute the command on the peer unit.

To send a command to a failover peer, perform the following steps:

**Step 1**
If you are in multiple context mode, use the `changeto` command to change to the context you want to configure. You cannot change contexts on the failover peer with the `failover exec` command.

If you are in single context mode, skip to the next step.

**Step 2**
Use the following command to send commands to the specified failover unit:

```
hostname(config)# failover exec {active | mate | standby}
```

Use the `active` or `standby` keyword to cause the command to be executed on the specified unit, even if that unit is the current unit. Use the `mate` keyword to cause the command to be executed on the failover peer.
Commands that cause a command mode change do not change the prompt for the current session. You must use the `show failover exec` command to display the command mode the command is executed in. See Changing Command Modes, page 51-22, for more information.

### Changing Command Modes

The `failover exec` command maintains a command mode state that is separate from the command mode of your terminal session. By default, the `failover exec` command mode starts in global configuration mode for the specified device. You can change that command mode by sending the appropriate command (such as the `interface` command) using the `failover exec` command. The session prompt does not change when you change mode using `failover exec`.

For example, if you are logged in to global configuration mode of the active unit of a failover pair, and you use the `failover exec active` command to change to interface configuration mode, the terminal prompt remains in global configuration mode, but commands entered using `failover exec` are entered in interface configuration mode.

The following examples shows the difference between the terminal session mode and the `failover exec` command mode. In the example, the administrator changes the `failover exec` mode on the active unit to interface configuration mode for the interface GigabitEthernet0/1. After that, all commands entered using `failover exec active` are sent to interface configuration mode for interface GigabitEthernet0/1. The administrator then uses failover exec active to assign an IP address to that interface. Although the prompt indicates global configuration mode, the `failover exec active` mode is in interface configuration mode.

```plaintext
hostname(config)# failover exec active interface GigabitEthernet0/1
hostname(config)# failover exec active ip address 192.168.1.1 255.255.255.0 standby 192.168.1.2
hostname(config)# router rip
hostname(config-router)#
```

Changing commands modes for your current session to the device does not affect the command mode used by the `failover exec` command. For example, if you are in interface configuration mode on the active unit, and you have not changed the `failover exec` command mode, the following command would be executed in global configuration mode. The result would be that your session to the device remains in interface configuration mode, while commands entered using `failover exec active` are sent to router configuration mode for the specified routing process.

```plaintext
hostname(config-if)# failover exec active router ospf 100
hostname(config-if)#
```

Use the `show failover exec` command to display the command mode on the specified device in which commands sent with the `failover exec` command are executed. The `show failover exec` command takes the same keywords as the `failover exec` command: `active`, `mate`, or `standby`. The `failover exec` mode for each device is tracked separately.

For example, the following is sample output from the `show failover exec` command entered on the standby unit:

```plaintext
hostname(config)# failover exec active interface GigabitEthernet0/1
hostname(config)# sh failover exec active
Active unit Failover EXEC is at interface sub-command mode

hostname(config)# sh failover exec standby
Standby unit Failover EXEC is at config mode

hostname(config)# sh failover exec mate
Active unit Failover EXEC is at interface sub-command mode
```
Security Considerations

The failover exec command uses the failover link to send commands to and receive the output of the command execution from the peer unit. You should use the failover key command to encrypt the failover link to prevent eavesdropping or man-in-the-middle attacks.

Limitations of Remote Command Execution

When you use remote commands you face the following limitations:

- If you upgrade one unit using the zero-downtime upgrade procedure and not the other, both units must be running software that supports the failover exec command for the command to work.
- Command completion and context help is not available for the commands in the cmd_string argument.
- In multiple context mode, you can only send commands to the peer context on the peer unit. To send commands to a different context, you must first change to that context on the unit to which you are logged in.
- You cannot use the following commands with the failover exec command:
  - changeto
  - debug (undebug)
- If the standby unit is in the failed state, it can still receive commands from the failover exec command if the failure is due to a service card failure; otherwise, the remote command execution will fail.
- You cannot use the failover exec command to switch from privileged EXEC mode to global configuration mode on the failover peer. For example, if the current unit is in privileged EXEC mode, and you enter failover exec mate configure terminal, the show failover exec mate output will show that the failover exec session is in global configuration mode. However, entering configuration commands for the peer unit using failover exec will fail until you enter global configuration mode on the current unit.
- You cannot enter recursive failover exec commands, such as failover exec mate failover exec mate command.
- Commands that require user input or confirmation must use the /nonconfirm option.

Controlling Failover

This sections describes how to control and monitor failover. This section includes the following topics:

- Forcing Failover, page 51-24
- Disabling Failover, page 51-24
- Restoring a Failed Unit or Failover Group, page 51-24
Forcing Failover

Enter the following command in the system execution space of the unit where the failover group is in the standby state:

```
hostname# failover active group group_id
```

Or, enter the following command in the system execution space of the unit where the failover group is in the active state:

```
hostname# no failover active group group_id
```

Entering the following command in the system execution space causes all failover groups to become active:

```
hostname# failover active
```

Disabling Failover

Disabling failover on an Active/Active failover pair causes the failover groups to remain in the active state on whichever unit they are active, no matter which unit they are configured to prefer. Enter the `no failover` command in the system execution space.

To disable failover, enter the following command:

```
hostname(config)# no failover
```

Restoring a Failed Unit or Failover Group

Restoring a failed unit or failover group moves the unit or failover group from the failed state to the standby state; it does not automatically make the failover group or unit active. Restored units or groups remain in the standby state until made active by failover (forced or natural). An exception is a failover group configured with failover preemption. If previously active, a failover group becomes active if it is configured with preemption and if the unit on which it failed is the preferred unit.

To restore a failed unit to an unfailed state, enter the following command:

```
hostname(config)# failover reset
```

To restore a failed Active/Active failover group to an unfailed state, enter the following command:

```
hostname(config)# failover reset group group_id
```

Testing the Failover Functionality

To test failover functionality, perform the following steps:

**Step 1**  Test that your active unit or failover group is passing traffic as expected by using FTP (for example) to send a file between hosts on different interfaces.

**Step 2**  Force a failover to the standby unit by entering the following command on the unit where the failover group containing the interface connecting your hosts is active:
hostname(config)# no failover active group group_id

Step 3 Use FTP to send another file between the same two hosts.
Step 4 If the test was not successful, enter the show failover command to check the failover status.
Step 5 When you are finished, you can restore the unit or failover group to active status by enter the following command on the unit where the failover group containing the interface connecting your hosts is active:

hostname(config)# failover active group group_id

Monitoring Active/Active Failover

To monitor Active/Active Failover, perform one of the following tasks. Commands are entered in the system execution space unless otherwise noted.

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show failover</td>
<td>Displays information about the failover state of the unit.</td>
</tr>
<tr>
<td>show failover group</td>
<td>Displays information about the failover state of the failover group. The information displayed is similar to that of the show failover command but limited to the specified group.</td>
</tr>
<tr>
<td>show monitor-interface</td>
<td>Displays information about the monitored interface. Enter this command within a security context.</td>
</tr>
<tr>
<td>show running-config failover</td>
<td>Displays the failover commands in the running configuration.</td>
</tr>
</tbody>
</table>

For more information about the output of the monitoring commands, see the Cisco ASA 5500 Series Command Reference.

Feature History for Active/Active Failover

Table 51-3 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active/Active failover</td>
<td>7.0</td>
<td>In an Active/Active failover configuration, both ASASMs can pass network traffic. We introduced this feature and the relevant commands.</td>
</tr>
<tr>
<td>IPv6 Support in failover</td>
<td>8.2(2)</td>
<td>We modified the following commands: failover interface ip, show failover, ipv6 address, show monitor-interface.</td>
</tr>
</tbody>
</table>
PART 14

Configuring Logging, SNMP, and Smart Call Home
Configuring Logging

This chapter describes how to configure and manage logs for the ASASM/ASASM and includes the following sections:

- Information About Logging, page 52-1
- Licensing Requirements for Logging, page 52-5
- Prerequisites for Logging, page 52-5
- Guidelines and Limitations, page 52-5
- Configuring Logging, page 52-6
- Monitoring the Logs, page 52-19
- Configuration Examples for Logging, page 52-20
- Feature History for Logging, page 52-20

Information About Logging

System logging is a method of collecting messages from devices to a server running a syslog daemon. Logging to a central syslog server helps in aggregation of logs and alerts. Cisco devices can send their log messages to a UNIX-style syslog service. A syslog service accepts messages and stores them in files, or prints them according to a simple configuration file. This form of logging provides protected long-term storage for logs. Logs are useful both in routine troubleshooting and in incident handling.

The ASASM system logs provide you with information for monitoring and troubleshooting the ASASM. With the logging feature, you can do the following:

- Specify which syslog messages should be logged.
- Disable or change the severity level of a syslog message.
- Specify one or more locations where syslog messages should be sent, including an internal buffer, one or more syslog servers, ASDM, an SNMP management station, specified e-mail addresses, or to Telnet and SSH sessions.
- Configure and manage syslog messages in groups, such as by severity level or class of message.
- Specify whether or not a rate-limit is applied to syslog generation.
- Specify what happens to the contents of the internal log buffer when it becomes full: overwrite the buffer, send the buffer contents to an FTP server, or save the contents to internal flash memory.
- Filter syslog messages by locations, severity level, class, or a custom message list.
This section includes the following topics:

- Logging in Multiple Context Mode, page 52-2
- Analyzing Syslog Messages, page 52-2
- Syslog Message Format, page 52-3
- Severity Levels, page 52-3
- Message Classes and Range of Syslog IDs, page 52-4
- Filtering Syslog Messages, page 52-4
- Using Custom Message Lists, page 52-4

Logging in Multiple Context Mode

Each security context includes its own logging configuration and generates its own messages. If you log in to the system or admin context, and then change to another context, messages you view in your session are only those messages that are related to the current context.

Syslog messages that are generated in the system execution space, including failover messages, are viewed in the admin context along with messages generated in the admin context. You cannot configure logging or view any logging information in the system execution space.

You can configure the ASASM/ASASM to include the context name with each message, which helps you differentiate context messages that are sent to a single syslog server. This feature also helps you to determine which messages are from the admin context and which are from the system; messages that originate in the system execution space use a device ID of system, and messages that originate in the admin context use the name of the admin context as the device ID.

Analyzing Syslog Messages

The following are some examples of the type of information you can obtain from a review of various syslog messages:

- Connections that are allowed by ASASM/ASASM security policies. These messages help you spot holes that remain open in your security policies.
- Connections that are denied by ASASM/ASASM security policies. These messages show what types of activity are being directed toward your secured inside network.
- Using the ACE deny rate logging feature shows attacks that are occurring on your ASASM/ASASM.
- IDS activity messages can show attacks that have occurred.
- User authentication and command usage provide an audit trail of security policy changes.
- Bandwidth usage messages show each connection that was built and torn down as well as the duration and traffic volume used.
- Protocol usage messages show the protocols and port numbers used for each connection.
- Address translation audit trail messages record NAT or PAT connections being built or torn down, which are useful if you receive a report of malicious activity coming from inside your network to the outside world.
Syslog Message Format

Syslog messages begin with a percent sign (%) and are structured as follows:

%ASA Level Message_number: Message_text

Field descriptions are as follows:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASASM</td>
<td>The syslog message facility code for messages that are generated by the ASASM/ASASM. This value is always ASA.</td>
</tr>
<tr>
<td>Level</td>
<td>1 through 7. The level reflects the severity of the condition described by the syslog message—the lower the number, the more severe the condition. See Table 52-1 for more information.</td>
</tr>
<tr>
<td>Message_number</td>
<td>A unique six-digit number that identifies the syslog message.</td>
</tr>
<tr>
<td>Message_text</td>
<td>A text string that describes the condition. This portion of the syslog message sometimes includes IP addresses, port numbers, or usernames.</td>
</tr>
</tbody>
</table>

Severity Levels

Table 52-1 lists the syslog message severity levels. You can assign custom colors to each of the severity levels to make it easier to distinguish them in the ASDM log viewers. To configure syslog message color settings, either choose the Tools > Preferences > Syslog tab or, in the log viewer itself, click Color Settings on the toolbar.

<table>
<thead>
<tr>
<th>Level Number</th>
<th>Severity Level</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>emergencies</td>
<td>System is unusable.</td>
</tr>
<tr>
<td>1</td>
<td>alert</td>
<td>Immediate action is needed.</td>
</tr>
<tr>
<td>2</td>
<td>critical</td>
<td>Critical conditions.</td>
</tr>
<tr>
<td>3</td>
<td>error</td>
<td>Error conditions.</td>
</tr>
<tr>
<td>4</td>
<td>warning</td>
<td>Warning conditions.</td>
</tr>
<tr>
<td>5</td>
<td>notification</td>
<td>Normal but significant conditions.</td>
</tr>
<tr>
<td>6</td>
<td>informational</td>
<td>Informational messages only.</td>
</tr>
<tr>
<td>7</td>
<td>debugging</td>
<td>Debugging messages only.</td>
</tr>
</tbody>
</table>

Note

The ASASM/ASASM does not generate syslog messages with a severity level of zero (emergencies). This level is provided in the logging command for compatibility with the UNIX syslog feature but is not used by the ASASM.
Message Classes and Range of Syslog IDs

For a list of syslog message classes and the ranges of syslog message IDs that are associated with each class, see the syslog messages guide.

Filtering Syslog Messages

You can filter generated syslog messages so that only certain syslog messages are sent to a particular output destination. For example, you could configure the ASASM/ASASM to send all syslog messages to one output destination and to send a subset of those syslog messages to a different output destination.

Specifically, you can configure the ASASM/ASASM so that syslog messages are directed to an output destination according to the following criteria:

- Syslog message ID number
- Syslog message severity level
- Syslog message class (equivalent to a functional area of the ASASM/ASASM)

You customize these criteria by creating a message list that you can specify when you set the output destination. Alternatively, you can configure the ASASM/ASASM to send a particular message class to each type of output destination independently of the message list.

You can use syslog message classes in two ways:

- Specify an output location for an entire category of syslog messages using the **logging class** command.
- Create a message list that specifies the message class using the **logging list** command.

The syslog message class provides a method of categorizing syslog messages by type, equivalent to a feature or function of the ASASM/ASASM. For example, the vpnc class denotes the VPN client.

All syslog messages in a particular class share the same initial three digits in their syslog message ID numbers. For example, all syslog message IDs that begin with the digits 611 are associated with the vpnc (VPN client) class. Syslog messages associated with the VPN client feature range from 611101 to 611323.

In addition, most of the ISAKMP syslog messages have a common set of prepended objects to help identify the tunnel. These objects precede the descriptive text of a syslog message when available. If the object is not known at the time that the syslog message is generated, the specific **heading = value** combination does not appear.

The objects are prefixed as follows:

- Group = `groupname`
- Username = `user`
- IP = `IP_address`

Where the group is the tunnel-group, the username is the username from the local database or AAA server, and the IP address is the public IP address of the remote access client or L2L peer.

Using Custom Message Lists

Creating a custom message list is a flexible way to exercise control over which syslog messages are sent to which output destination. In a custom syslog message list, you specify groups of syslog messages using any or all of the following criteria: severity level, message IDs, ranges of syslog message IDs, or message class.

For example, you can use message lists to do the following:
• Select syslog messages with the severity levels of 1 and 2 and send them to one or more e-mail addresses.
• Select all syslog messages associated with a message class (such as ha) and save them to the internal buffer.

A message list can include multiple criteria for selecting messages. However, you must add each message selection criterion with a new command entry. It is possible to create a message list that includes overlapping message selection criteria. If two criteria in a message list select the same message, the message is logged only once.

## Licensing Requirements for Logging

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

### Prerequisites for Logging

Logging has the following prerequisites:

• The syslog server must run a server program called syslogd. Windows (except for Windows 95 and Windows 98) provides a syslog server as part of its operating system. For Windows 95 and Windows 98, you must obtain a syslogd server from another vendor.

• To view logs generated by the ASASM/ASASM, you must specify a logging output destination. If you enable logging without specifying a logging output destination, the ASASM/ASASM generates messages but does not save them to a location from which you can view them. You must specify each different logging output destination separately. For example, to designate more than one syslog server as an output destination, enter a new command for each syslog server.

### Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

#### Context Mode Guidelines

Supported in single and multiple context modes.

#### Firewall Mode Guidelines

Supported in routed and transparent firewall modes.

#### IPv6 Guidelines

Does not support IPv6.

#### Additional Guidelines

• Sending syslogs over TCP is not supported on a standby ASASM.
The ASASM supports the configuration of 16 syslog servers with the `logging host` command in single context mode. In multiple context mode, the limitation is 4 servers per context.

### Configuring Logging

This section describes how to configure logging and includes the following topics:

- Enabling Logging, page 52-6
- Configuring an Output Destination, page 52-6

**Note**
The minimum configuration depends on what you want to do and what your requirements are for handling syslog messages in the ASASM/ASASM.

### Enabling Logging

To enable logging, enter the following command:

```
logging enable
```

**Example:**
```
hostname(config)# logging enable
```

**Command** | **Purpose**
--- | ---
`logging enable` | Enables logging. To disable logging, enter the `no logging enable` command.

**What to Do Next**

See the “Configuring an Output Destination” section on page 52-6.

### Configuring an Output Destination

To optimize syslog message usage for troubleshooting and performance monitoring, we recommend that you specify one or more locations where syslog messages should be sent, including an internal log buffer, one or more external syslog servers, ASDM, an SNMP management station, the console port, specified e-mail addresses, or Telnet and SSH sessions.

This section includes the following topics:

- Sending Syslog Messages to an External Syslog Server, page 52-8
- Sending Syslog Messages to the Internal Log Buffer, page 52-9
- Sending Syslog Messages to an E-mail Address, page 52-10
- Sending Syslog Messages to ASDM, page 52-11
- Sending Syslog Messages to the Console Port, page 52-11
- Sending Syslog Messages to an SNMP Server, page 52-12
- Sending Syslog Messages to a Telnet or SSH Session, page 52-12
- Creating a Custom Event List, page 52-13
• Generating Syslog Messages in EMBLEM Format to a Syslog Server, page 52-14
• Generating Syslog Messages in EMBLEM Format to Other Output Destinations, page 52-14
• Changing the Amount of Internal Flash Memory Available for Logs, page 52-15
• Configuring the Logging Queue, page 52-15
• Sending All Syslog Messages in a Class to a Specified Output Destination, page 52-16
• Enabling Secure Logging, page 52-16
• Including the Device ID in Non-EMBLEM Format Syslog Messages, page 52-17
• Including the Date and Time in Syslog Messages, page 52-18
• Disabling a Syslog Message, page 52-18
• Changing the Severity Level of a Syslog Message, page 52-18
• Limiting the Rate of Syslog Message Generation, page 52-19
Sending Syslog Messages to an External Syslog Server

You can archive messages according to the available disk space on the external syslog server, and manipulate logging data after it is saved. For example, you could specify actions to be executed when certain types of syslog messages are logged, extract data from the log and save the records to another file for reporting, or track statistics using a site-specific script.

To send syslog messages to an external syslog server, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Configures the ASASM/ASASM to send messages to a syslog server.</td>
</tr>
<tr>
<td>`logging host interface_name syslog_ip [tcp[/port]</td>
<td>udp[/port] [format emblem]]`</td>
</tr>
<tr>
<td>Example: <code>hostname(config)# logging host dmz1 192.168.1.5 udp 1026 format emblem</code></td>
<td>You can configure the ASASM to send data to a syslog server using either UDP or TCP, but not both. The default protocol is UDP if you do not specify a protocol.</td>
</tr>
<tr>
<td>Step 2</td>
<td>Specifies which syslog messages should be sent to the syslog server. You can specify the severity level number (1 through 7) or name. For example, if you set the severity level to 3, then the ASASM/ASASM send syslog messages for severity levels 3, 2, and 1. You can specify a custom message list that identifies the syslog messages to send to the syslog server.</td>
</tr>
<tr>
<td>`logging trap {severity_level</td>
<td>message_list}`</td>
</tr>
<tr>
<td>Example: <code>hostname(config)# logging trap errors</code></td>
<td></td>
</tr>
</tbody>
</table>

Example: `hostname(config)# logging host dmz1 192.168.1.5 udp 1026 format emblem`
## Configuring Logging

### Sending Syslog Messages to the Internal Log Buffer

To send syslog messages to the internal log buffer, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1**

logging buffered *(severity_level | message_list)*

Example:

hostname(config)# logging buffered critical
hostname(config)# logging buffered level 2
hostname(config)# logging buffered notif-list

| **Step 2**

logging buffer-size *bytes*

Example:

hostname(config)# logging buffer-size 16384

| **Step 3**

Choose one of the following options:

logging flash-bufferwrap

Example:

hostname(config)# logging flash-bufferwrap

| **Step 3**

logging permit-hostdown

Example:

hostname(config)# logging permit-hostdown

(Optional) Disables the feature to block new connections when a TCP-connected syslog server is down. If the ASASM/ASASM is configured to send syslog messages to a TCP-based syslog server, and if either the syslog server is down or the log queue is full, then new connections are blocked. New connections are allowed again after the syslog server is back up and the log queue is no longer full. For more information about the log queue, see the “Configuring the Logging Queue” section on page 52-15.

(Optional) Sets the logging facility to a value other than 20, which is what most UNIX systems expect.

(Optional) Specifies which syslog messages should be sent to the internal log buffer, which serves as a temporary storage location. New messages are appended to the end of the list. When the buffer is full, that is, when the buffer wraps, old messages are overwritten as new messages are generated, unless you configure the ASASM/ASASM to save the full buffer to another location. To empty the internal log buffer, enter the clear logging buffer command.

Changes the size of the internal log buffer. The buffer size is 4 KB.

When saving the buffer content to another location, the ASASM/ASASM create log files with names that use the following time-stamp format:

`LOG-YYYY-MM-DD-HHMMSS.TXT`

where `YYYY` is the year, `MM` is the month, `DD` is the day of the month, and `HHMMSS` is the time in hours, minutes, and seconds.

The ASASM/ASASM continues to save new messages to the internal log buffer and saves the full log buffer content to the internal flash memory.
Configuring Logging

Sending Syslog Messages to an E-mail Address

To send syslog messages to an e-mail address, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td></td>
</tr>
<tr>
<td>logging mail {severity_level</td>
<td>message_list}</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# logging mail high-priority</td>
</tr>
<tr>
<td>Step 2</td>
<td></td>
</tr>
<tr>
<td>logging from-address email_address</td>
<td>Specifies the source e-mail address to be used when sending syslog messages to an e-mail address.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# logging from-address <a href="mailto:xxx-001@example.com">xxx-001@example.com</a></td>
</tr>
</tbody>
</table>
Configuring Logging

Sending Syslog Messages to ASDM

To send syslog messages to ASDM, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> logging asdm (severity_level</td>
<td>message_list)</td>
</tr>
<tr>
<td>Example: hostname(config)# logging asdm 2</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> logging asdm-buffer-size num_of_msgs</td>
<td>Specifies the number of syslog messages to be retained in the ASDM log buffer. To empty the current content of the ASDM log buffer, enter the clear logging asdm command.</td>
</tr>
<tr>
<td>Example: hostname(config)# logging asdm-buffer-size 200</td>
<td></td>
</tr>
</tbody>
</table>

Sending Syslog Messages to the Console Port

To send syslog messages to the console port, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>logging console (severity_level</td>
<td>message_list)</td>
</tr>
<tr>
<td>Example: hostname(config)# logging console errors</td>
<td></td>
</tr>
</tbody>
</table>
### Sending Syslog Messages to an SNMP Server

To enable logging to an SNMP server, enter the following command:

```
logging history [logging_list | level]
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`logging history [logging_list</td>
<td>level]`</td>
</tr>
</tbody>
</table>

**Example:**
```
hostname(config)# logging history errors
```

### Sending Syslog Messages to a Telnet or SSH Session

To send syslog messages to a Telnet or SSH session, perform the following steps:

**Step 1**

```
logging monitor (severity_level | message_list)
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`logging monitor (severity_level</td>
<td>message_list)`</td>
</tr>
</tbody>
</table>

**Example:**
```
hostname(config)# logging monitor 6
```

**Step 2**

```
terminal monitor
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>terminal monitor</code></td>
<td>Enables logging to the current session only. If you log out and then log in again, you need to reenter this command. To disable logging to the current session, enter the <code>terminal no monitor</code> command.</td>
</tr>
</tbody>
</table>

**Example:**
```
hostname(config)# terminal monitor
```
## Creating a Custom Event List

To create a custom event list, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1**
logging list name {level level [class message_class] | message start_id[-end_id])

Example:
hostname(config)# logging list notif-list level 3

Specifies criteria for selecting messages to be saved in the internal log buffer. For example, if you set the severity level to 3, then the ASASM sends syslog messages for severity levels 3, 2, and 1.

The name argument specifies the name of the list. The level level keyword and argument pair specify the severity level. The class message_class keyword and argument pair specify a particular message class. The message start_id[-end_id] keyword and argument pair specify an individual syslog message number or a range of numbers.

**Note**
Do not use the names of severity levels as the name of a syslog message list. Prohibited names include emergencies, alert, critical, error, warning, notification, informational, and debugging. Similarly, do not use the first three characters of these words at the beginning of an event list name. For example, do not use an event list name that starts with the characters err.

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 2**
logging list name {level level [class message_class] | message start_id[-end_id])

Example:
hostname(config)# logging list notif-list message 104024-105999
hostname(config)# logging list notif-list level critical
hostname(config)# logging list notif-list level warning class ha

(Optional) Adds more criteria for message selection to the list. Enter the same command as in the previous step, specifying the name of the existing message list and the additional criterion. Enter a new command for each criterion that you want to add to the list. For example, you can specify criteria for syslog messages to be included in the list as the following:

- Syslog message IDs that fall into the range of 104024 to 105999.
- All syslog messages with the critical severity level or higher (emergency, alert, or critical).
- All ha class syslog messages with the warning severity level or higher (emergency, alert, critical, error, or warning).

**Note**
A syslog message is logged if it satisfies any of these conditions. If a syslog message satisfies more than one of the conditions, the message is logged only once.
### Generating Syslog Messages in EMBLEM Format to a Syslog Server

To generate syslog messages in EMBLEM format to a syslog server, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`logging host interface_name ip_address {tcp[/port]</td>
<td>udp[/port]} [format emblem]`</td>
</tr>
<tr>
<td>Example: <code>hostname(config)# logging host interface_1 127.0.0.1 udp format emblem</code></td>
<td></td>
</tr>
</tbody>
</table>

### Generating Syslog Messages in EMBLEM Format to Other Output Destinations

To generate syslog messages in EMBLEM format to other output destinations, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>logging emblem</code></td>
<td>Sends syslog messages in EMBLEM format to output destinations other than a syslog server, such as Telnet or SSH sessions.</td>
</tr>
<tr>
<td>Example: <code>hostname(config)# logging emblem</code></td>
<td></td>
</tr>
</tbody>
</table>
Changing the Amount of Internal Flash Memory Available for Logs

To change the amount of internal flash memory available for logs, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><strong>logging flash-maximum-allocation kbytes</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><strong>hostname(config)# logging flash-maximum-allocation 1200</strong></td>
</tr>
<tr>
<td>Specifies the maximum amount of internal flash memory available for saving log files. By default, the ASASM can use up to 1 MB of internal flash memory for log data. The minimum amount of internal flash memory that must be free for the ASASM/ASASM to save log data is 3 MB. If a log file being saved to internal flash memory would cause the amount of free internal flash memory to fall below the configured minimum limit, the ASASM/ASASM deletes the oldest log files to ensure that the minimum amount of memory remains free after saving the new log file. If there are no files to delete or if, after all old files have been deleted, free memory is still below the limit, the ASASM/ASASM fails to save the new log file.</td>
<td></td>
</tr>
</tbody>
</table>

| **Step 2** | **logging flash-minimum-free kbytes** |
| **Example:** | **hostname(config)# logging flash-minimum-free 4000** |
| Specifies the minimum amount of internal flash memory that must be free for the ASASM/ASASM to save a log file. |

Configuring the Logging Queue

To configure the logging queue, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **logging queue message_count** | Specifies the number of syslog messages that the ASASM/ASASM can hold in its queue before sending them to the configured output destination. The ASASM/ASASM has a fixed number of blocks in memory that can be allocated for buffering syslog messages while they are waiting to be sent to the configured output destination. The number of blocks required depends on the length of the syslog message queue and the number of syslog servers specified. The default queue size is 512 syslog messages. The queue size is limited only by block memory availability. Valid values are from 0 to 8192 messages, depending on the platform. If the logging queue is set to zero, the queue is the maximum configurable size (8192 messages), depending on the platform. The maximum queue size by platform is as follows:  
  * ASA-5505—1024  
  * ASA-5510—2048  
  * On all other platforms—8192 |
| **Example:** | **hostname(config)# logging queue 300** |
# Sending All Syslog Messages in a Class to a Specified Output Destination

To send all syslog messages in a class to a specified output destination, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`logging class message_class {buffered</td>
<td>console</td>
</tr>
</tbody>
</table>

**Example:**
```
hostname(config)# logging class ha buffered alerts
```

---

# Enabling Secure Logging

To enable secure logging, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`logging host interface_name syslog_ip [tcp/port</td>
<td>udp/port] [format emblem] [secure]`</td>
</tr>
</tbody>
</table>

**Note** Secure logging does not support UDP; an error occurs if you try to use this protocol.

**Example:**
```
hostname(config)# logging host inside 10.0.0.1 TCP/1500 secure
```
Including the Device ID in Non-EMBLEM Format Syslog Messages

To include the device ID in non-EMBLEM format syslog messages, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| `logging device-id [context-name | hostname | ipaddress interface_name | string text]` | Configures the ASASM/ASASM to include a device ID in non-EMBLEM-format syslog messages. You can specify only one type of device ID for syslog messages. The `context-name` keyword indicates that the name of the current context should be used as the device ID (applies to multiple context mode only). If you enable the logging device ID for the admin context in multiple context mode, messages that originate in the system execution space use a device ID of `system`, and messages that originate in the admin context use the name of the admin context as the device ID. The `hostname` keyword specifies that the hostname of the ASASM should be used as the device ID. The `ipaddress` `interface_name` keyword and argument pair specify that the interface IP address specified as `interface_name` should be used as the device ID. If you use the `ipaddress` keyword, the device ID becomes the specified ASASM interface IP address, regardless of the interface from which the syslog message is sent. This keyword provides a single, consistent device ID for all syslog messages that are sent from the device. The `string` `text` keyword and argument pair specify that the text string should be used as the device ID. The string can include as many as 16 characters. You cannot use blank spaces or any of the following characters:
- `&` (ampersand)
- `'` (single quote)
- `"` (double quote)
- `<` (less than)
- `>` (greater than)
- `?` (question mark)

**Note** If enabled, the device ID does not appear in EMBLEM-formatted syslog messages nor in SNMP traps.

Example:

```
hostname(config)# logging device-id hostname
hostname(config)# logging device-id context-name
```
Including the Date and Time in Syslog Messages

To include the date and time in syslog messages, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>logging timestamp</strong></td>
<td>Specifies that syslog messages should include the date and time that they were generated. To remove the date and time from syslog messages, enter the <strong>no logging timestamp</strong> command.</td>
</tr>
</tbody>
</table>

Example:

```
hostname(config)# logging timestamp
LOG-2008-10-24-081856.TXT
```

Disabling a Syslog Message

To disable a specified syslog message, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>no logging message</strong> message_number</td>
<td>Prevents the ASASM/ASASM from generating a particular syslog message. To reenable a disabled syslog message, enter the <strong>logging message</strong> message_number command (for example, <strong>logging message 113019</strong>). To reenable logging of all disabled syslog messages, enter the <strong>clear config logging disabled</strong> command.</td>
</tr>
</tbody>
</table>

Example:

```
hostname(config)# no logging message 113019
```

Changing the Severity Level of a Syslog Message

To change the severity level of a syslog message, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>logging message</strong> message_ID level severity_level**</td>
<td>Specifies the severity level of a syslog message. To reset the severity level of a syslog message to its setting, enter the <strong>no logging message</strong> message_ID level current_severity_level command (for example, <strong>no logging message 113019 level 5</strong>). To reset the severity level of all modified syslog messages to their settings, enter the <strong>clear configure logging level</strong> command.</td>
</tr>
</tbody>
</table>

Example:

```
hostname(config)# logging message 113019 level 5
```
Limiting the Rate of Syslog Message Generation

To limit the rate of syslog message generation, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>logging rate-limit {unlimited</td>
<td>{num [interval]}} message syslog_id</td>
</tr>
</tbody>
</table>

**Example:**
hostname(config)# logging rate-limit 1000 600 level 6

---

Monitoring the Logs

To monitor the logs and assist in monitoring the system performance, enter one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show logging</td>
<td>Shows syslog messages, including the severity level.</td>
</tr>
<tr>
<td>show logging message</td>
<td>Shows a list of syslog messages with modified severity levels and disabled syslog messages.</td>
</tr>
<tr>
<td>show logging message message_ID</td>
<td>Shows the severity level of a specific syslog message.</td>
</tr>
<tr>
<td>show logging queue</td>
<td>Shows the logging queue and queue statistics.</td>
</tr>
<tr>
<td>show logging rate-limit</td>
<td>Shows the disallowed syslog messages.</td>
</tr>
<tr>
<td>show running-config logging rate-limit</td>
<td>Shows the current logging rate-limit setting.</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows the logging information that displays for the `show logging` command:

hostname(config)# show logging
Syslog logging: enabled
   Facility: 16
   Timestamp logging: disabled
   Standby logging: disabled
   Deny Conn when Queue Full: disabled
   Console logging: disabled
   Monitor logging: disabled
   Buffer logging: disabled
   Trap logging: level errors, facility 16, 3607 messages logged
   Logging to infrastructure 10.1.2.3
History logging: disabled
Device ID: 'inside' interface IP address "10.1.1.1"
Mail logging: disabled
ASDM logging: disabled

Configuration Examples for Logging

The following examples show how to control both whether a syslog message is enabled and the severity level of the specified syslog message:

```
hostname(config)# show logging message 403503
syslog 403503: -level errors (enabled)

hostname(config)# logging message 403503 level 1
hostname(config)# show logging message 403503
syslog 403503: -level errors, current-level alerts (enabled)

hostname(config)# no logging message 403503
hostname(config)# show logging message 403503
syslog 403503: -level errors, current-level alerts (disabled)

hostname(config)# logging message 403503
hostname(config)# show logging message 403503
syslog 403503: -level errors, current-level alerts (enabled)

hostname(config)# no logging message 403503 level 3
hostname(config)# show logging message 403503
syslog 403503: -level errors (enabled)
```

Feature History for Logging

Table 52-2 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logging</td>
<td>7.0(1)</td>
<td>Provides ASASM network logging information through various output destinations, and includes the option to view and save log files.</td>
</tr>
<tr>
<td>Rate limit</td>
<td>7.0(4)</td>
<td>Limits the rate at which syslog messages are generated. We introduced the following command: <code>logging rate-limit</code>.</td>
</tr>
<tr>
<td>Logging list</td>
<td>7.2(1)</td>
<td>Creates a logging list to use in other commands to specify messages by various criteria (logging level, event class, and message IDs). We introduced the following command: <code>logging list</code>.</td>
</tr>
</tbody>
</table>
Table 52-2  Feature History for Logging (continued)

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Secure logging</td>
<td>8.0(2)</td>
<td>Specifies that the connection to the remote logging host should use SSL/TLS. This option is valid only if the protocol selected is TCP.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>We modified the following command: <code>logging host</code>.</td>
</tr>
<tr>
<td>Logging class</td>
<td>8.0(4), 8.1(1)</td>
<td>Added support for the ipaa event class of logging messages.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>We modified the following command: <code>logging class</code>.</td>
</tr>
<tr>
<td>Logging class and saved logging</td>
<td>8.2(1)</td>
<td>Added support for the dap event class of logging messages.</td>
</tr>
<tr>
<td>buffers</td>
<td></td>
<td>We modified the following command: <code>logging class</code>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Added support to clear the saved logging buffers (ASDM, internal, FTP, and flash).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>We introduced the following command: <code>clear logging queue bufferwrap</code>.</td>
</tr>
<tr>
<td>Password encryption</td>
<td>8.3(1)</td>
<td>Added support for password encryption.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>We modified the following command: <code>logging ftp server</code>.</td>
</tr>
<tr>
<td>Enhanced logging and connection</td>
<td>8.3(2)</td>
<td>When you configure a syslog server to use TCP, and the syslog server is unavailable, the ASASM blocks new connections that generate syslog messages</td>
</tr>
<tr>
<td>blocking</td>
<td></td>
<td>until the server becomes available again (for example, VPN, firewall, and cut-through-proxy connections). This feature has been enhanced to also block</td>
</tr>
<tr>
<td></td>
<td></td>
<td>new connections when the logging queue on the ASASM is full; connections resume when the logging queue is cleared.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This feature was added for compliance with Common Criteria EAL4+. Unless required, we recommended allowing connections when syslog messages cannot</td>
</tr>
<tr>
<td></td>
<td></td>
<td>be sent or received. To allow connections, continue to use the <code>logging permit-hostdown</code> command.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>We modified the following command: <code>show logging</code>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>We introduced the following syslog messages: 414005, 414006, 414007, and 414008.</td>
</tr>
</tbody>
</table>
CHAPTER 53

Configuring NetFlow Secure Event Logging (NSEL)

This chapter describes how to configure NSEL, a security logging mechanism that is built on NetFlow Version 9 technology, and how to handle events and syslog messages through NSEL.

This chapter includes the following sections:

- Information About NSEL, page 53-1
- Licensing Requirements for NSEL, page 53-3
- Prerequisites for NSEL, page 53-3
- Guidelines and Limitations, page 53-4
- Configuring NSEL, page 53-4
- Monitoring NSEL, page 53-10
- Configuration Examples for NSEL, page 53-12
- Where to Go Next, page 53-13
- Additional References, page 53-13
- Feature History for NSEL, page 53-14

Information About NSEL

The ASASM and ASASM support NetFlow Version 9 services. For more information about NetFlow services, see the “RFCs” section on page 53-14.

The ASASM and ASASM implementations of NSEL provide a stateful, IP flow tracking method that exports records that indicate significant events in a flow. In stateful flow tracking, tracked flows go through a series of state changes. NSEL events are used to export data about flow status and are triggered by the event that caused the state change.

The significant events that are tracked include flow-create, flow-teardown, flow-denied (excluding those flows that are denied by EtherType ACLs), and flow-update. In addition, the ASA and ASASM implementation of NSEL generates periodic NSEL events and flow-update events to provide periodic byte counters over the duration of the flow. These events are usually time-driven, which makes them more in line with traditional Netflow; however, these events may also be triggered by state changes in the flow.

Each NSEL record has an event ID and an extended event ID field, which describes the flow event.

The ASASM and ASASM implementations of NSEL provide the following major functions:
- Tracks flow-create, flow-teardown, and flow-denied events, and generates appropriate NSEL data records.
- Triggers flow-update events and generates appropriate NSEL data records.
- Defines and exports templates that describe the progression of a flow. Templates describe the format of the data records that are exported through NetFlow. Each event has several record formats or templates associated with it.
- Tracks configured NSEL collectors and delivers templates and data records to these configured NSEL collectors through NetFlow over UDP only.
- Sends template information periodically to NSEL collectors. Collectors receive template definitions, normally before receiving flow records.
- Filters NSEL events based on the traffic and event type through Modular Policy Framework, then sends records to different collectors. Traffic is matched based on the order in which classes are configured. After a match is found, no other classes are checked. The supported event types are flow-create, flow-denied, flow-teardown, flow-update, and all. Records can be sent to different collectors. For example, with two collectors, you can do the following:
  - Log all flow-denied events that match access list 1 to collector 1.
  - Log all flow-create events to collector 1.
  - Log all flow-teardown events to collector 2.
  - Log all flow-update events to collector 1.
- Delays the export of flow-create events.

Using NSEL and Syslog Messages

Table 53-1 lists the syslog messages that have an equivalent NSEL event, event ID, and extended event ID. The extended event ID provides more detail about the event (for example, which ACL—ingress or egress—has denied a flow).

Note

Enabling NetFlow to export flow information makes the syslog messages that are listed in Table 53-1 redundant. In the interest of performance, we recommend that you disable redundant syslog messages, because the same information is exported through NetFlow. You can enable or disable individual syslog messages by following the procedure in the “Disabling and Reenabling NetFlow-related Syslog Messages” section on page 53-9.

**Table 53-1  Syslog Messages and Equivalent NSEL Events**

<table>
<thead>
<tr>
<th>Syslog Message</th>
<th>Description</th>
<th>NSEL Event ID</th>
<th>NSEL Extended Event ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>106100</td>
<td>Generated whenever an ACL is encountered.</td>
<td>1—Flow was created (if the ACL allowed the flow).</td>
<td>0—If the ACL allowed the flow.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3—Flow was denied (if the ACL denied the flow).</td>
<td>1001—Flow was denied by the ingress ACL.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1002—Flow was denied by the egress ACL.</td>
</tr>
<tr>
<td>106015</td>
<td>A TCP flow was denied because the first packet was not a SYN packet.</td>
<td>3—Flow was denied.</td>
<td>1004—Flow was denied because the first packet was not a TCP SYN packet.</td>
</tr>
</tbody>
</table>
Table 53-1  Syslog Messages and Equivalent NSEL Events (continued)

<table>
<thead>
<tr>
<th>Syslog Message</th>
<th>Description</th>
<th>NSEL Event ID</th>
<th>NSEL Extended Event ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>106023</td>
<td>When a flow was denied by an ACL attached to an interface through the access-group command.</td>
<td>3—Flow was denied.</td>
<td>1001—Flow was denied by the ingress ACL. 1002—Flow was denied by the egress ACL.</td>
</tr>
<tr>
<td>302013, 302015, 302017, 302020</td>
<td>TCP, UDP, GRE, and ICMP connection creation.</td>
<td>1—Flow was created.</td>
<td>0—Ignore.</td>
</tr>
<tr>
<td>302014, 302016, 302018, 302021</td>
<td>TCP, UDP, GRE, and ICMP connection teardown.</td>
<td>2—Flow was deleted.</td>
<td>0—Ignore. &gt; 2000—Flow was torn down.</td>
</tr>
<tr>
<td>313001</td>
<td>An ICMP packet to the device was denied.</td>
<td>3—Flow was denied.</td>
<td>1003—To-the-box flow was denied because of configuration.</td>
</tr>
<tr>
<td>313008</td>
<td>An ICMP v6 packet to the device was denied.</td>
<td>3—Flow was denied.</td>
<td>1003—To-the-box flow was denied because of configuration.</td>
</tr>
<tr>
<td>710003</td>
<td>An attempt to connect to the device interface was denied.</td>
<td>3—Flow was denied.</td>
<td>1003—To-the-box flow was denied because of configuration.</td>
</tr>
</tbody>
</table>

**Note**
When NSEL and syslog messages are both enabled, there is no guarantee of chronological ordering between the two logging types.

**Licensing Requirements for NSEL**

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

**Prerequisites for NSEL**

NSEL has the following prerequisites:

- IP address and hostname assignments must be unique throughout the NetFlow configuration.
- You must have at least one configured collector before you can use NSEL.
- You must configure NSEL collectors before you can configure filters via Modular Policy Framework.
Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

Context Mode Guidelines
Supported in single and multiple context mode.

Firewall Mode Guidelines
Supported in routed and transparent firewall mode.

IPv6 Guidelines
Supports IPv6 for the class-map, match any and class-default commands. The match access-list commands only support IPv4 access lists.

Additional Guidelines and Limitations
- If you have previously configured flow-export actions using the flow-export enable command, and you upgrade to a later version, then your configuration is automatically converted to the new Modular Policy Framework flow-export event-type command, which is described under the policy-map command.
- Flow-export actions are not supported in interface-based policies. You can configure flow-export actions in a class-map only with the match access-list, match any, or class-default commands. You can only apply flow-export actions in a global service policy.
- To view bandwidth usage for NetFlow records (not available in real-time), you must use the threat detection feature.

Configuring NSEL

This section describes how to configure NSEL and includes the following topics:
- Configuring NSEL Collectors, page 53-5
- Configuring Template Timeout Intervals, page 53-7
- Changing the Time Interval for Sending Flow-Update Events to a Collector, page 53-8
- Disabling and Reenabling NetFlow-related Syslog Messages, page 53-9
- Clearing Runtime Counters, page 53-10
Configuring NSEL Collectors

To configure NSEL collectors, enter the following command:

```
flow-export destination interface-name
ipv4-address|hostname udp-port
```

**Example:**
```
hostname (config)# flow-export destination inside 209.165.200.225 2002
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`flow-export destination interface-name ipv4-address</td>
<td>hostname udp-port`</td>
</tr>
</tbody>
</table>

**Note** Make sure that collector applications use the Event Time field to correlate events.

What to Do Next

See the “Configuring Flow-Export Actions Through Modular Policy Framework” section on page 53-5.

Configuring Flow-Export Actions Through Modular Policy Framework

To export NSEL events by defining all classes with flow-export actions, perform the following steps:

**Step 1**
```
class-map flow_export_class
```

**Example:**
```
hostname (config-pmap)# class-map flow_export_class
```

**Purpose** Defines the class map that identifies traffic for which NSEL events need to be exported. The `flow_export_class` argument is the name of the class map.

**Step 2**
Choose one of the following options:

**match access-list flow_export_acl**

**Example:**
```
hostname (config-cmap)# match access-list flow_export_acl
```

**match any**

**Example:**
```
hostname (config-cmap)# match any
```

**Purpose** Configures the access list to match specific traffic. The `flow_export_acl` argument is the name of the access list. Matches any traffic.
### Configuring Ne tFlow Secure Event Logging (NSEL)

**Step 3**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>policy-map flow_export_policy</td>
<td>Defines the policy map to apply flow-export actions to the defined classes. The <em>flow_export_policy</em> argument is the name of the policy map. If you create a new policy map and apply it globally according to Step 6, the remaining inspection policies are deactivated. Alternatively, to insert a NetFlow class in the existing policy, enter the <code>class flow_export_class</code> command after the <code>policy-map global_policy</code> command. For more information about creating or modifying the Modular Policy Framework, see Chapter 30, “Configuring a Service Policy Using the Modular Policy Framework.”</td>
</tr>
</tbody>
</table>

**Example:**

```plaintext
hostname(config)# policy-map flow_export_policy
```

**Step 4**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>class flow_export_class</td>
<td>Defines the class to apply flow-export actions. The <em>flow_export_class</em> argument is the name of the class.</td>
</tr>
</tbody>
</table>

**Example:**

```plaintext
hostname (config-pmap)# class flow_export_class
```

**Step 5**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>flow-export event-type event-type destination flow_export_host1 [flow_export_host2]</td>
<td>Configures a flow-export action. The <em>event_type</em> keyword is the name of the supported event being filtered. The <em>flow_export_host</em> argument is the IP address of a host. The <em>destination</em> keyword is the IP address of the configured collector.</td>
</tr>
</tbody>
</table>

**Example:**

```plaintext
hostname (config-pmap-c)# flow-export event-type all destination 209.165.200.230
```

**Step 6**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>service-policy flow_export_policy global</td>
<td>Adds or edits the service policy globally. The <em>flow_export_policy</em> argument is the name of the policy map.</td>
</tr>
</tbody>
</table>

**Example:**

```plaintext
hostname (config)# service-policy flow_export_policy global
```
What to Do Next

See the “Configuring Template Timeout Intervals” section on page 53-7.

Configuring Template Timeout Intervals

To configure template timeout intervals, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>flow-export template timeout-rate minutes</code></td>
<td>Specifies the interval at which template records are sent to all configured output destinations. The <code>template</code> keyword indicates the template-specific configurations. The <code>timeout-rate</code> keyword specifies the time before templates are resent. The <code>minutes</code> argument specifies the time interval in minutes at which the templates are resent. The default value is 30 minutes.</td>
</tr>
</tbody>
</table>

Example:

```
hostname (config)# flow-export template timeout-rate 15
```
What to Do Next

See the “Changing the Time Interval for Sending Flow-Update Events to a Collector” section on page 53-8.

Changing the Time Interval for Sending Flow-Update Events to a Collector

To change the time interval at which periodic flow-update events are to be sent to a collector, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>flow-export active refresh-interval value</td>
<td>Configures NetFlow parameters for active connections. The value argument specifies the time interval between flow-update events in minutes. Valid values are from 1 - 60 minutes. The default value is 1 minute. If you have already configured the flow-export delay flow-create command, and you then configure the flow-export active refresh-interval command with an interval value that is not at least 5 seconds more than the delay value, the following warning message appears at the console: WARNING: The current delay flow-create value configuration may cause flow-update events to appear before flow-creation events.</td>
</tr>
<tr>
<td>Example: hostname (config)# flow-export active refresh-interval 30</td>
<td></td>
</tr>
</tbody>
</table>

If you have already configured the flow-export active refresh-interval command, and you then configure the flow-export delay flow-create command with a delay value that is not at least 5 seconds less than the interval value, the following warning message appears at the console: WARNING: The current delay flow-create value configuration may cause flow-update events to appear before flow-creation events.
Chapter 53      Configuring NetFlow Secure Event Logging (NSEL)

What to Do Next

See the “Delaying Flow-Create Events” section on page 53-9.

Delaying Flow-Create Events

To delay the sending of flow-create events, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>flow-export delay flow-create seconds</code></td>
<td>Delays the sending of a flow-create event by the specified number of</td>
</tr>
<tr>
<td></td>
<td>seconds. The <code>seconds</code> argument indicates the amount of time allowed</td>
</tr>
<tr>
<td></td>
<td>for the delay in seconds. If this command is not configured, there is</td>
</tr>
<tr>
<td></td>
<td>no delay, and the flow-create event is exported as soon as the flow is</td>
</tr>
<tr>
<td></td>
<td>created. If the flow is torn down before the configured delay, the</td>
</tr>
<tr>
<td></td>
<td>flow-create event is not sent; an extended flow teardown event is sent</td>
</tr>
<tr>
<td>Example:</td>
<td>instead.</td>
</tr>
<tr>
<td><code>hostname (config)# flow-export delay flow-create 10</code></td>
<td></td>
</tr>
</tbody>
</table>

What to Do Next


Disabling and Reenabling NetFlow-related Syslog Messages

To disable and reenable NetFlow-related syslog messages, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td><code>logging flow-export-syslogs disable</code></td>
<td>Disables syslog messages that have become redundant because of NSEL.</td>
</tr>
<tr>
<td><strong>Note</strong></td>
<td>Although you execute this command in global configuration mode, it is</td>
</tr>
<tr>
<td></td>
<td>not stored in the configuration. Only the <code>no logging</code> message <code>xxxxxx</code></td>
</tr>
<tr>
<td></td>
<td>commands are stored in the configuration.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>hostname(config)# logging flow-export-syslogs disable</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td><code>logging message xxxxxx</code></td>
<td>Reenables syslog messages individually, where <code>xxxxxx</code> is the specified</td>
</tr>
<tr>
<td></td>
<td>syslog message that you want to reenable.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>hostname(config)# logging message 302013</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td></td>
</tr>
<tr>
<td><code>logging flow-export-syslogs enable</code></td>
<td>Reenables all NSEL events at the same time.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>hostname(config)# logging flow-export-syslogs enable</code></td>
<td></td>
</tr>
</tbody>
</table>
Monitoring NSEL

You can use syslog messages to help troubleshoot errors or monitor system usage and performance. You can view real-time syslog messages that have been saved in the log buffer in a separate window, which include an explanation of the message, details about the message, and recommended actions to take, if necessary, to resolve an error. For more information, see the “Using NSEL and Syslog Messages” section on page 53-2.

NSEL Monitoring Commands

To monitor NSEL, enter one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show flow-export counters</td>
<td>Shows runtime counters, including statistical data and error data, for NSEL.</td>
</tr>
<tr>
<td>show logging flow-export-syslogs</td>
<td>Lists all syslog messages that are captured by NSEL events.</td>
</tr>
<tr>
<td>show running-config flow-export</td>
<td>Shows the currently configured NetFlow commands.</td>
</tr>
<tr>
<td>show running-config logging</td>
<td>Shows disabled syslog messages, which are redundant syslog messages, because they export the same information through NetFlow.</td>
</tr>
</tbody>
</table>

Examples

The following example shows how to display flow-export counters:

```
hostname (config)# show flow-export counters
```
destination: inside 209.165.200.225 2055

Statistics:
  packets sent  250

Errors:
  block allocation errors  0
  invalid interface  0
  template send failure  0
  no route to collector  0

The following example shows how to display the flow-export active configuration:

hostname (config)# show running-config flow-export active
flow-export active refresh-interval 2

The following example shows how to display the flow-export delay configuration:

hostname (config)# show running-config flow-export delay
flow-export delay flow-create 30

The following example shows how to display the flow-export destination configurations:

hostname (config)# show running-config flow-export destination
flow-export destination inside 192.68.10.70 9996

The following example shows how to display the flow-export template configuration:

hostname (config)# show running-config flow-export template
flow-export template timeout-rate 1

The following example shows how to display flow-export syslog messages:

hostname# show logging flow-export-syslogs

<table>
<thead>
<tr>
<th>Syslog ID</th>
<th>Type</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>302013</td>
<td>Flow Created</td>
<td>Enabled</td>
</tr>
<tr>
<td>302015</td>
<td>Flow Created</td>
<td>Enabled</td>
</tr>
<tr>
<td>302017</td>
<td>Flow Created</td>
<td>Enabled</td>
</tr>
<tr>
<td>302020</td>
<td>Flow Created</td>
<td>Enabled</td>
</tr>
<tr>
<td>302014</td>
<td>Flow Deleted</td>
<td>Enabled</td>
</tr>
<tr>
<td>302016</td>
<td>Flow Deleted</td>
<td>Enabled</td>
</tr>
<tr>
<td>302018</td>
<td>Flow Deleted</td>
<td>Enabled</td>
</tr>
<tr>
<td>302021</td>
<td>Flow Deleted</td>
<td>Enabled</td>
</tr>
<tr>
<td>106015</td>
<td>Flow Denied</td>
<td>Enabled</td>
</tr>
<tr>
<td>106023</td>
<td>Flow Denied</td>
<td>Enabled</td>
</tr>
<tr>
<td>313001</td>
<td>Flow Denied</td>
<td>Enabled</td>
</tr>
<tr>
<td>313008</td>
<td>Flow Denied</td>
<td>Enabled</td>
</tr>
<tr>
<td>710003</td>
<td>Flow Denied</td>
<td>Enabled</td>
</tr>
<tr>
<td>106100</td>
<td>Flow Created/Denied</td>
<td>Enabled</td>
</tr>
</tbody>
</table>

The following example shows how to display current syslog message settings:

hostname (config)# show running-config logging

no logging message 313008
no logging message 313001
### Configuration Examples for NSEL

The following examples show how to filter NSEL events, with the specified collectors already configured:

- **flow-export destination inside 209.165.200.2055**
- **flow-export destination outside 209.165.201.29 2055**
- **flow-export destination outside 209.165.201.27 2055**

Log all events between hosts 209.165.200.224 and hosts 209.165.201.224 to 209.165.200.230, and log all other events to 209.165.201.29:

```
hostname (config)# access-list flow_export_acl permit ip host 209.165.200.224 host 209.165.201.224
hostname (config)# class-map flow_export_class
hostname (config-cmap)# match access-list flow_export_acl
hostname (config)# policy-map flow_export_policy
hostname (config-pmap)# class flow_export_class
hostname (config-pmap-c)# flow-export event-type all destination 209.165.200.230
hostname (config-pmap-c)# class class-default
hostname (config-pmap-c)# flow-export event-type all destination 209.165.201.29
hostname (config)# service-policy flow_export_policy global
```

Log flow-create events to 209.165.200.230, flow-teardown events to 209.165.201.29, flow-denied events to 209.165.201.27, and flow-update events to 209.165.200.230:

```
hostname (config)# policy-map flow_export_policy
hostname (config-pmap)# class class-default
hostname (config-pmap-c)# flow-export event-type flow-creation destination 209.165.200.230
hostname (config-pmap-c)# flow-export event-type flow-teardown destination 209.165.201.29
hostname (config-pmap-c)# flow-export event-type flow-denied destination 209.165.201.27
hostname (config-pmap-c)# flow-export event-type flow-update destination 209.165.200.230
hostname (config)# service-policy flow_export_policy global
```

Log flow-create events between hosts 209.165.200.224 and 209.165.200.230 to 209.165.201.29, and log all flow-denied events to 209.165.201.27:

```
hostname (config)# access-list flow_export_acl permit ip host 209.165.200.224 host 209.165.200.230
hostname (config)# class-map flow_export_class
hostname (config-cmap)# match access-list flow_export_acl
hostname (config)# policy-map flow_export_policy
hostname (config-pmap)# class flow_export_class
hostname (config-pmap-c)# flow-export event-type flow-creation destination 209.165.200.230
hostname (config-pmap-c)# flow-export event-type flow-denied destination 209.165.201.29
hostname (config-pmap-c)# class class-default
hostname (config-pmap-c)# flow-export event-type flow-denied destination 209.165.201.27
hostname (config)# service-policy flow_export_policy global
```

**Note**

You must enter the following command:

```
hostname (config-pmap-c)# flow-export event-type flow-denied destination 209.165.201.27
```

for `flow_export_acl`, because traffic is not checked after the first match, and you must explicitly define the action to log flow-denied events that match `flow_export_acl`.

Log all traffic except traffic between hosts 209.165.201.27 and 209.165.201.50 to 209.165.201.27:

```
hostname (config)# access-list flow_export_acl deny ip host 209.165.201.30 host 209.165.201.50
```
hostname (config)# access-list flow_export_acl permit ip any any
hostname (config)# class-map flow_export_class
hostname (config-cmap)# match access-list flow_export_acl
hostname (config)# policy-map flow_export_policy
hostname (config-pmap)# class flow_export_class
hostname (config-pmap-c)# flow-export event-type all destination 209.165.201.27
hostname (config)# service-policy flow_export_policy global

Where to Go Next

To configure the syslog server, see Chapter 52, “Configuring Logging.”

Additional References

For additional information related to implementing NSEL, see the following sections:

- Related Documents, page 53-14
- RFCs, page 53-14
Chapter 53  Configuring NetFlow Secure Event Logging (NSEL)

Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Using NSEL and Syslog Messages, page 53-2</td>
<td>syslog messages guide</td>
</tr>
<tr>
<td>Information about the implementation of NSEL on the ASASM and ASASM</td>
<td>Cisco ASA 5500 Series Implementation Note for NetFlow Collectors</td>
</tr>
<tr>
<td></td>
<td>See the following article at</td>
</tr>
<tr>
<td>Configuring NetFlow on the ASASM and ASASM using ASDM</td>
<td>See the following article at</td>
</tr>
</tbody>
</table>

RFCs

<table>
<thead>
<tr>
<th>RFC</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>3954</td>
<td>Cisco Systems NetFlow Services Export Version 9</td>
</tr>
</tbody>
</table>

Feature History for NSEL

Table 53-2 lists each feature change and the platform release in which it was implemented..
Table 53-2  Feature History for NSEL

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>NetFlow</td>
<td>8.1(1)</td>
<td>The NetFlow feature enhances the ASASM logging capabilities by logging flow-based events through the NetFlow protocol. NetFlow Version 9 services are used to export information about the progression of a flow from start to finish. The NetFlow implementation exports records that indicate significant events in the life of a flow. This implementation is different from traditional NetFlow, which exports data about flows at regular intervals. The NetFlow module also exports records about flows that are denied by access lists. You can configure an ASA 5580 to send the following events using NetFlow: flow create, flow teardown, and flow denied (only flows denied by ACLs are reported). We introduced the following commands: clear flow-export counters, flow-export enable, flow-export destination, flow-export template timeout-rate, logging flow-export syslogs enable, logging flow-export syslogs disable, show flow-export counters, show logging flow-export-syslogs.</td>
</tr>
<tr>
<td>NetFlow Filtering</td>
<td>8.1(2)</td>
<td>You can filter NetFlow events based on traffic and event type, then send records to different collectors. For example, you can log all flow-create events to one collector, and log flow-denied events to a different collector. We modified the following commands: class, class-map, flow-export event-type destination, match access-list, policy-map, service-policy. For short-lived flows, NetFlow collectors benefit from processing a single event instead of two events: flow create and flow teardown. You can configure a delay before sending the flow-create event. If the flow is torn down before the timer expires, only the flow teardown event is sent. The teardown event includes all information regarding the flow; no loss of information occurs. We introduced the following command: flow-export delay flow-create.</td>
</tr>
<tr>
<td>NSEL</td>
<td>8.2(1)</td>
<td>The NetFlow feature has been ported to all available models of the ASASM.</td>
</tr>
<tr>
<td>NSEL</td>
<td>8.4(5)</td>
<td>Flow-update events have been introduced to provide periodic byte counters for flow traffic. You can change the time interval at which flow-update events are sent to the NetFlow collector. You can filter to which collectors flow-update records will be sent. We introduced the following command: flow-export active refresh-interval. We modified the following command: flow-export event-type.</td>
</tr>
</tbody>
</table>
Chapter 54

Configuring SNMP

This chapter describes how to configure SNMP to monitor the ASASM/ASASM and includes the following sections:

- Information About SNMP, page 54-1
- Licensing Requirements for SNMP, page 54-16
- Prerequisites for SNMP, page 54-16
- Guidelines and Limitations, page 54-16
- Configuring SNMP, page 54-17
- Troubleshooting Tips, page 54-23
- Monitoring SNMP, page 54-25
- Configuration Examples for SNMP, page 54-27
- Where to Go Next, page 54-28
- Additional References, page 54-28
- Feature History for SNMP, page 54-30

Information About SNMP

SNMP is an application-layer protocol that facilitates the exchange of management information between network devices and is part of the TCP/IP protocol suite. This section describes SNMP and includes the following topics:

- Information About SNMP Terminology, page 54-2
- Information About MIBs and Traps, page 54-3
- SNMP Object Identifiers, page 54-3
- SNMP Physical Vendor Type Values, page 54-5
- Supported Tables in MIBs, page 54-11
- Supported Traps (Notifications), page 54-12
- SNMP Version 3, page 54-14

The ASASM/ASASM provides support for network monitoring using SNMP Versions 1, 2c, and 3, and supports the use of all three versions simultaneously. The SNMP agent running on the ASASM interface lets you monitor the ASASM and through network management systems (NMSs), such as HP OpenView.
The ASASM/ASASM supports SNMP read-only access through issuance of a GET request. SNMP write access is not allowed, so you cannot make changes with SNMP. In addition, the SNMP SET request is not supported.

You can configure the ASASM/ASASM to send traps, which are unsolicited messages from the managed device to the management station for certain events (event notifications) to an NMS, or you can use the NMS to browse the MIBs on the ASASM. MIBs are a collection of definitions, and the ASASM/ASASM maintains a database of values for each definition. Browsing a MIB means issuing a series of GET-NEXT or GET-BULK requests of the MIB tree from the NMS to determine values.

The ASASM/ASASM has an SNMP agent that notifies designated management stations if events occur that are predefined to require a notification, for example, when a link in the network goes up or down. The notification it sends includes an SNMP OID, which identifies itself to the management stations. The ASASM/ASASM SNMP agent also replies when a management station asks for information.

### Information About SNMP Terminology

Table 54-1 lists the terms that are commonly used when working with SNMP:

<table>
<thead>
<tr>
<th>Term</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent</td>
<td>The SNMP server running on the ASASM. The SNMP agent has the following features:</td>
</tr>
<tr>
<td></td>
<td>• Responds to requests for information and actions from the network management station.</td>
</tr>
<tr>
<td></td>
<td>• Controls access to its Management Information Base, the collection of objects that the SNMP manager can view or change.</td>
</tr>
<tr>
<td></td>
<td>• Does not allow set operations.</td>
</tr>
<tr>
<td>Browsing</td>
<td>Monitoring the health of a device from the network management station by polling required information from the SNMP agent on the device. This activity may include issuing a series of GET-NEXT or GET-BULK requests of the MIB tree from the network management station to determine values.</td>
</tr>
<tr>
<td>Management Information Bases (MIBs)</td>
<td>Standardized data structures for collecting information about packets, connections, buffers, failovers, and so on. MIBs are defined by the product, protocols, and hardware standards used by most network devices. SNMP network management stations can browse MIBs and request specific data or events be sent as they occur.</td>
</tr>
<tr>
<td>Network management stations (NMSs)</td>
<td>The PCs or workstations set up to monitor SNMP events and manage devices, such as the ASASM/ASASM.</td>
</tr>
<tr>
<td>Object identifier (OID)</td>
<td>The system that identifies a device to its NMS and indicates to users the source of information monitored and displayed.</td>
</tr>
<tr>
<td>Trap</td>
<td>Predefined events that generate a message from the SNMP agent to the NMS. Events include alarm conditions such as linkup, linkdown, coldstart, warmstart, authentication, or syslog messages.</td>
</tr>
</tbody>
</table>
Information About MIBs and Traps

MIBs are either standard or enterprise-specific. Standard MIBs are created by the IETF and documented in various RFCs. A trap reports significant events occurring on a network device, most often errors or failures. SNMP traps are defined in either standard or enterprise-specific MIBs. Standard traps are created by the IETF and documented in various RFCs. SNMP traps are compiled into the ASASM/ASASM software.

If needed, you can also download RFCs, standard MIBs, and standard traps from the following locations:

http://www.ietf.org/

Download a complete list of Cisco MIBs, traps, and OIDs from the following location:


In addition, download Cisco OIDs by FTP from the following location:

Note

In software versions 7.2(1), 8.0(2), and later, the interface information accessed through SNMP refreshes about every 5 seconds. As a result, we recommend that you wait for at least 5 seconds between consecutive polls.

SNMP Object Identifiers

Each Cisco system-level product has an SNMP object identifier (OID) for use as a MIB-II sysObjectID. The CISCO-PRODUCTS-MIB includes the OIDs that can be reported in the sysObjectID object in the SNMPv2-MIB. You can use this value to identify the model type. Table 54-2 lists the sysObjectID OIDs for ASASM models.

Table 54-2 SNMP Object Identifiers

<table>
<thead>
<tr>
<th>Product Identifier</th>
<th>sysObjectID</th>
<th>Model Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASA 5505</td>
<td>ciscoASA5505 (ciscoProducts 745)</td>
<td>Cisco ASA 5505</td>
</tr>
<tr>
<td>ASA 5510</td>
<td>ciscoASA5510 (ciscoProducts 669)</td>
<td>Cisco ASA 5510</td>
</tr>
<tr>
<td>ASA 5510</td>
<td>ciscoASA5510sc (ciscoProducts 773)</td>
<td>Cisco ASA 5510 security context</td>
</tr>
<tr>
<td>ASA 5510</td>
<td>ciscoASA5510sy (ciscoProducts 774)</td>
<td>Cisco ASA 5510 system context</td>
</tr>
<tr>
<td>ASA 5520</td>
<td>ciscoASA5520 (ciscoProducts 670)</td>
<td>Cisco ASA 5520</td>
</tr>
<tr>
<td>ASA 5520</td>
<td>ciscoASA5520sc (ciscoProducts 671)</td>
<td>Cisco ASA 5520 security context</td>
</tr>
<tr>
<td>ASA 5520</td>
<td>ciscoASA5520sy (ciscoProducts 764)</td>
<td>Cisco ASA 5520 system context</td>
</tr>
<tr>
<td>ASA 5540</td>
<td>ciscoASA5540 (ciscoProducts 672)</td>
<td>Cisco ASA 5540</td>
</tr>
<tr>
<td>ASA 5540</td>
<td>ciscoASA5540sc (ciscoProducts 673)</td>
<td>Cisco ASA 5540 security context</td>
</tr>
<tr>
<td>ASA 5540</td>
<td>ciscoASA5540sy (ciscoProducts 765)</td>
<td>Cisco ASA 5540 system context</td>
</tr>
<tr>
<td>ASA 5550</td>
<td>ciscoASA5550 (ciscoProducts 753)</td>
<td>Cisco ASA 5550</td>
</tr>
<tr>
<td>ASA 5550</td>
<td>ciscoASA5550sc (ciscoProducts 763)</td>
<td>Cisco ASA 5550 security context</td>
</tr>
<tr>
<td>ASA 5550</td>
<td>ciscoASA 5550sy (ciscoProducts 766)</td>
<td>Cisco ASA 5550 system context</td>
</tr>
</tbody>
</table>
### Table 54-2  SNMP Object Identifiers (continued)

| ASA5580 | ciscoASA5580 (ciscoProducts 914) | Cisco ASA 5580 |
| ASA5580 | ciscoASA5580 (ciscoProducts 915) | Cisco ASA 5580 security context |
| ASA5580 | ciscoASA5580 (ciscoProducts 916) | Cisco ASA 5580 system context |
| ASA5585-SSP10 | ciscoASA5585Ssp10 (ciscoProducts 1194) | ASA 5585-X SSP-10 |
| ASA5585-SSP20 | ciscoASA5585Ssp20 (ciscoProducts 1195) | ASA 5585-X SSP-20 |
| ASA5585-SSP40 | ciscoASA5585Ssp40 (ciscoProducts 1196) | ASA 5585-X SSP-40 |
| ASA5585-SSP60 | ciscoASA5585Ssp60 (ciscoProducts 1197) | ASA 5585-X SSP-60 |
| ASA5585-SSP10 | ciscoASA5585Ssp10sc (ciscoProducts 1198) | ASA 5585-X SSP-10 security context |
| ASA5585-SSP20 | ciscoASA5585Ssp20sc (ciscoProducts 1199) | ASA 5585-X SSP-20 security context |
| ASA5585-SSP40 | ciscoASA5585Ssp40sc (ciscoProducts 1200) | ASA 5585-X SSP-40 security context |
| ASA5585-SSP60 | ciscoASA5585Ssp60sc (ciscoProducts 1201) | ASA 5585-X SSP-60 security context |
| ASA5585-SSP10 | ciscoASA5585Ssp10sy (ciscoProducts 1202) | ASA 5585-X SSP-10 system context |
| ASA5585-SSP20 | ciscoASA5585Ssp20sy (ciscoProducts 1203) | ASA 5585-X SSP-20 system context |
| ASA5585-SSP40 | ciscoASA5585Ssp40sy (ciscoProducts 1204) | ASA 5585-X SSP-40 system context |
| ASA5585-SSP60 | ciscoASA5585Ssp60sy (ciscoProducts 1205) | ASA 5585-X SSP-60 system context |
| ASA Services Module for Catalyst switches | ciscoAsaSm1 (ciscoProducts 1277) | Adaptive Security Appliance (ASASM) Services Module for Catalyst switches |
| ASA Services Module for Catalyst switches security context | ciscoAsaSm1sc (ciscoProducts 1275) | Adaptive Security Appliance (ASASM) Services Module for Catalyst switches security context |
| ASA Services Module for Catalyst switches security context with No Payload Encryption | ciscoAsaSm1K7sc (ciscoProducts 1334) | Adaptive Security Appliance (ASASM) Services Module for Catalyst switches security context with No Payload Encryption |
| ASA Services Module for Catalyst switches system context | ciscoAsaSm1sy (ciscoProducts 1276) | Adaptive Security Appliance (ASASM) Services Module for Catalyst switches system context |
| ASA Services Module for Catalyst switches system context with No Payload Encryption | ciscoAsaSm1K7sy (ciscoProducts 1335) | Adaptive Security Appliance (ASASM) Services Module for Catalyst switches system context with No Payload Encryption |
| ASA Services Module for Catalyst switches system context with No Payload Encryption | ciscoAsaSm1K7 (ciscoProducts 1336) | Adaptive Security Appliance (ASASM) Services Module for Catalyst switches with No Payload Encryption |
| ASASM 5512 | ciscoASA5512 (ciscoProducts 1407) | ASASM 5512 Adaptive Security Appliance |
| ASASM 5525 | ciscoASA5525 (ciscoProducts 1408) | ASASM 5525 Adaptive Security Appliance |
| ASASM 5545 | ciscoASA5545 (ciscoProducts 1409) | ASASM 5545 Adaptive Security Appliance |
### SNMP Physical Vendor Type Values

Each Cisco chassis or standalone system has a unique type number for SNMP use. The `entPhysicalVendorType` OIDs are defined in the CISCO-ENTITY-VENDORTYPE-OID-MIB. This value is returned in the `entPhysicalVendorType` object from the ASASM/ASASM SNMP agent. You can use this value to identify the type of component (module, power supply, fan, sensors, CPU, and so on). Table 54-3 lists the physical vendor type values for the ASASM/ASASM models.

<table>
<thead>
<tr>
<th>Item</th>
<th><code>entPhysicalVendorType OID</code> Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASA Services Module for Catalyst switches</td>
<td>cevCat6kWsSvcAsaSm1 (cevModuleCat6000Type 169)</td>
</tr>
<tr>
<td>ASA Services Module for Catalyst switches with No Payload Encryption</td>
<td>cevCat6kWsSvcAsaSm1K7 (cevModuleCat6000Type 186)</td>
</tr>
<tr>
<td>ASA 5505 chassis</td>
<td>cevChassisASA5505 (cevChassis 560)</td>
</tr>
<tr>
<td>ASA 5510 chassis</td>
<td>cevChassisASA5510 (cevChassis 447)</td>
</tr>
<tr>
<td>Cisco Adaptive Security Appliance (ASA) 5512 Adaptive Security Appliance</td>
<td>cevChassisASA5512 (cevChassis 1113)</td>
</tr>
</tbody>
</table>
### Table 54-3 SNMP Physical Vendor Type Values (continued)

<table>
<thead>
<tr>
<th>Device Description</th>
<th>Vendor Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco Adaptive Security Appliance (ASA) 5512</td>
<td>cevChassisASA5512K7 (cevChassis 1108)</td>
</tr>
<tr>
<td>Adaptive Security Appliance with No Payload Encryption</td>
<td></td>
</tr>
<tr>
<td>Cisco Adaptive Security Appliance (ASA) 5515</td>
<td>cevChassisASA5515 (cevChassis 1114)</td>
</tr>
<tr>
<td>Adaptive Security Appliance</td>
<td></td>
</tr>
<tr>
<td>Cisco Adaptive Security Appliance (ASA) 5515</td>
<td>cevChassisASA5515K7 (cevChassis 1109)</td>
</tr>
<tr>
<td>Adaptive Security Appliance with No Payload Encryption</td>
<td></td>
</tr>
<tr>
<td>ASA 5520 chassis</td>
<td>cevChassisASA5520 (cevChassis 448)</td>
</tr>
<tr>
<td>Cisco Adaptive Security Appliance (ASA) 5525</td>
<td>cevChassisASA5525 (cevChassis 1115)</td>
</tr>
<tr>
<td>Adaptive Security Appliance</td>
<td></td>
</tr>
<tr>
<td>Cisco Adaptive Security Appliance (ASA) 5525</td>
<td>cevChassisASA5525K7 (cevChassis 1110)</td>
</tr>
<tr>
<td>Adaptive Security Appliance with No Payload Encryption</td>
<td></td>
</tr>
<tr>
<td>ASA 5540 chassis</td>
<td>cevChassisASA5540 (cevChassis 449)</td>
</tr>
<tr>
<td>Cisco Adaptive Security Appliance (ASA) 5545</td>
<td>cevChassisASA5545 (cevChassis 1116)</td>
</tr>
<tr>
<td>Adaptive Security Appliance</td>
<td></td>
</tr>
<tr>
<td>Cisco Adaptive Security Appliance (ASA) 5545</td>
<td>cevChassisASA5545K7 (cevChassis 1111)</td>
</tr>
<tr>
<td>Adaptive Security Appliance with No Payload Encryption</td>
<td></td>
</tr>
<tr>
<td>ASA 5550 chassis</td>
<td>cevChassisASA5550 (cevChassis 564)</td>
</tr>
<tr>
<td>Cisco Adaptive Security Appliance (ASA) 5555</td>
<td>cevChassisASA5555 (cevChassis 1117)</td>
</tr>
<tr>
<td>Adaptive Security Appliance</td>
<td></td>
</tr>
<tr>
<td>Cisco Adaptive Security Appliance (ASA) 5555</td>
<td>cevChassisASA5555K7 (cevChassis 1112)</td>
</tr>
<tr>
<td>Adaptive Security Appliance with No Payload Encryption</td>
<td></td>
</tr>
<tr>
<td>ASA 5580 chassis</td>
<td>cevChassisASA5580 (cevChassis 704)</td>
</tr>
<tr>
<td>Central Processing Unit for Cisco Adaptive Security Appliance 5512</td>
<td>cevCpuAsa5512 (cevModuleCpuType 229)</td>
</tr>
<tr>
<td>Central Processing Unit for Cisco Adaptive Security Appliance 5512 with no Payload Encryption</td>
<td>cevCpuAsa5512K7 (cevModuleCpuType 224)</td>
</tr>
<tr>
<td>Central Processing Unit for Cisco Adaptive Security Appliance 5515</td>
<td>cevCpuAsa5515 (cevModuleCpuType 230)</td>
</tr>
<tr>
<td>Central Processing Unit for Cisco Adaptive Security Appliance 5515 with no Payload Encryption</td>
<td>cevCpuAsa5515K7 (cevModuleCpuType 225)</td>
</tr>
<tr>
<td>Central Processing Unit for Cisco Adaptive Security Appliance 5525</td>
<td>cevCpuAsa5525 (cevModuleCpuType 231)</td>
</tr>
<tr>
<td>Central Processing Unit for Cisco Adaptive Security Appliance 5525 with no Payload Encryption</td>
<td>cevCpuAsa5525K7 (cevModuleCpuType 226)</td>
</tr>
<tr>
<td>Central Processing Unit for Cisco Adaptive Security Appliance 5545</td>
<td>cevCpuAsa5545 (cevModuleCpuType 232)</td>
</tr>
<tr>
<td>Central Processing Unit for Cisco Adaptive Security Appliance 5545 with no Payload Encryption</td>
<td>cevCpuAsa5545K7 (cevModuleCpuType 227)</td>
</tr>
</tbody>
</table>
### Table 54-3  SNMP Physical Vendor Type Values (continued)

<table>
<thead>
<tr>
<th>Central Processing Unit for Cisco Adaptive Security Appliance 5555</th>
<th>cevCpuAsa5555 (cevModuleCpuType 233)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Central Processing Unit for Cisco Adaptive Security Appliance 5555 with no Payload Encryption</td>
<td>cevCpuAsa5555K7 (cevModuleCpuType 228)</td>
</tr>
<tr>
<td>CPU for ASA 5580</td>
<td>cevCpuAsa5580 (cevModuleType 200)</td>
</tr>
<tr>
<td>CPU for ASA 5585 SSP-10</td>
<td>cevCpuAsa5585Ssp10 (cevModuleCpuType 204)</td>
</tr>
<tr>
<td>CPU for ASA 5585 SSP-10 No Payload Encryption</td>
<td>cevCpuAsa5585Ssp10K7 (cevModuleCpuType 205)</td>
</tr>
<tr>
<td>CPU for ASA 5585 SSP-20</td>
<td>cevCpuAsa5585Ssp20 (cevModuleCpuType 206)</td>
</tr>
<tr>
<td>CPU for ASA 5585 SSP-20 No Payload Encryption</td>
<td>cevCpuAsa5585Ssp20K7 (cevModuleCpuType 207)</td>
</tr>
<tr>
<td>CPU for ASA 5585 SSP-40</td>
<td>cevCpuAsa5585Ssp40 (cevModuleCpuType 208)</td>
</tr>
<tr>
<td>CPU for ASA 5585 SSP-40 No Payload Encryption</td>
<td>cevCpuAsa5585Ssp40K7 (cevModuleCpuType 209)</td>
</tr>
<tr>
<td>CPU for ASA 5585 SSP-60</td>
<td>cevCpuAsa5585Ssp60 (cevModuleCpuType 210)</td>
</tr>
<tr>
<td>CPU for ASA 5585 SSP-60 No Payload Encryption</td>
<td>cevCpuAsa5585Ssp60K (cevModuleCpuType 211)</td>
</tr>
<tr>
<td>CPU for Cisco ASA Services Module for Catalyst switches</td>
<td>cevCpuAsaSm1 (cevModuleCpuType 222)</td>
</tr>
<tr>
<td>CPU for Cisco ASA Services Module with No Payload Encryption for Catalyst switches</td>
<td>cevCpuAsaSm1K7 (cevModuleCpuType 223)</td>
</tr>
<tr>
<td>Chassis Cooling Fan in Adaptive Security Appliance 5512</td>
<td>cevFanASA5512ChassisFan (cevFan 163)</td>
</tr>
<tr>
<td>Chassis Cooling Fan in Adaptive Security Appliance 5512 with No Payload Encryption</td>
<td>cevFanASA5512K7ChassisFan (cevFan 172)</td>
</tr>
<tr>
<td>Chassis Cooling Fan in Adaptive Security Appliance 5515</td>
<td>cevFanASA5515ChassisFan (cevFan 164)</td>
</tr>
<tr>
<td>Chassis Cooling Fan in Adaptive Security Appliance 5515 with No Payload Encryption</td>
<td>cevFanASA5515K7ChassisFan (cevFan 171)</td>
</tr>
<tr>
<td>Chassis Cooling Fan in Adaptive Security Appliance 5525</td>
<td>cevFanASA5525ChassisFan (cevFan 165)</td>
</tr>
<tr>
<td>Chassis Cooling Fan in Adaptive Security Appliance 5525 with No Payload Encryption</td>
<td>cevFanASA5525K7ChassisFan (cevFan 170)</td>
</tr>
<tr>
<td>Chassis Cooling Fan in Adaptive Security Appliance 5545</td>
<td>cevFanASA5545ChassisFan (cevFan 166)</td>
</tr>
<tr>
<td>Chassis Cooling Fan in Adaptive Security Appliance 5545 with No Payload Encryption</td>
<td>cevFanASA5545K7ChassisFan (cevFan 169)</td>
</tr>
<tr>
<td>Power Supply Fan in Adaptive Security Appliance 5545</td>
<td>cevFanASA5545K7PSFan (cevFan 161)</td>
</tr>
<tr>
<td>Power Supply Fan in Adaptive Security Appliance 5545 with No Payload Encryption</td>
<td>cevFanASA5545PSFan (cevFan 159)</td>
</tr>
<tr>
<td>Chassis Cooling Fan in Adaptive Security Appliance 5555</td>
<td>cevFanASA5555ChassisFan (cevFan 167)</td>
</tr>
<tr>
<td>Chassis Cooling Fan in Adaptive Security Appliance 5555 with No Payload Encryption</td>
<td>cevFanASA5555K7ChassisFan (cevFan 168)</td>
</tr>
</tbody>
</table>
### Information About SNMP

#### Table 54-3  SNMP Physical Vendor Type Values (continued)

<table>
<thead>
<tr>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power Supply Fan in Adaptive Security Appliance 5555</td>
<td>cevFanASA5555PSFan (cevFan 160)</td>
</tr>
<tr>
<td>Power Supply Fan in Adaptive Security Appliance 5555 with No Payload Encryption</td>
<td>cevFanASA5555PSFanK7 (cevFan 162)</td>
</tr>
<tr>
<td>Fan type for ASA 5580</td>
<td>cevFanASA5580Fan (cevFan 138)</td>
</tr>
<tr>
<td>Power supply fan for ASA 5585-X</td>
<td>cevFanASA5585PSFan (cevFan 146)</td>
</tr>
<tr>
<td>ASA 5580 4-port GE copper interface card</td>
<td>cevModuleASA5580Pm4x1GeCu (cevModuleASA5580Type 1)</td>
</tr>
<tr>
<td>10-Gigabit Ethernet interface</td>
<td>cevPort10GigEthernet (cevPort 315)</td>
</tr>
<tr>
<td>Gigabit Ethernet port</td>
<td>cevPortGe (cevPort 109)</td>
</tr>
<tr>
<td>Power Supply unit in Adaptive Security Appliance 5545</td>
<td>cevPowerSupplyASA5545PSInput (cevPowerSupply 323)</td>
</tr>
<tr>
<td>Presence Sensor for Power Supply input in Adaptive Security Appliance 5545</td>
<td>cevPowerSupplyASA5545SPresence (cevPowerSupply 321)</td>
</tr>
<tr>
<td>Power Supply unit in Adaptive Security Appliance 5555</td>
<td>cevPowerSupplyASA5555PSInput (cevPowerSupply 324)</td>
</tr>
<tr>
<td>Presence Sensor for Power Supply input in Adaptive Security Appliance 5555</td>
<td>cevPowerSupplyASA5555SPresence (cevPowerSupply 322)</td>
</tr>
<tr>
<td>Power supply input for ASA 5580</td>
<td>cevPowerSupplyASA5580PSInput (cevPowerSupply 292)</td>
</tr>
<tr>
<td>Power supply input for ASA 5585</td>
<td>cevPowerSupplyASA5585PSInput (cevPowerSupply 304)</td>
</tr>
<tr>
<td>Cisco Adaptive Security Appliance (ASA) 5512 Chassis Fan sensor</td>
<td>cevSensorASA5512ChassisFanSensor (cevSensor 120)</td>
</tr>
<tr>
<td>Chassis Ambient Temperature Sensor for Cisco Adaptive Security Appliance 5512</td>
<td>cevSensorASA5512ChassisTemp (cevSensor 107)</td>
</tr>
<tr>
<td>Central Processing Unit Temperature Sensor for Cisco Adaptive Security Appliance 5512</td>
<td>cevSensorASA5512CPUTemp (cevSensor 96)</td>
</tr>
<tr>
<td>Cisco Adaptive Security Appliance (ASA) 5512 with No Payload Encryption Chassis Fan sensor</td>
<td>cevSensorASA5512K7ChassisFanSensor (cevSensor 125)</td>
</tr>
<tr>
<td>Central Processing Unit Temperature Sensor for Cisco Adaptive Security Appliance 5512 with No Payload Encryption</td>
<td>cevSensorASA5512K7CPUTemp (cevSensor 102)</td>
</tr>
<tr>
<td>Sensor for Chassis Cooling Fan in Adaptive Security Appliance 5512</td>
<td>cevSensorASA5512PSFanSensor (cevSensor 119)</td>
</tr>
<tr>
<td>Cisco Adaptive Security Appliance (ASA) 5515 Chassis Fan sensor</td>
<td>cevSensorASA5515ChassisFanSensor (cevSensor 121)</td>
</tr>
<tr>
<td>Chassis Ambient Temperature Sensor for Cisco Adaptive Security Appliance 5515</td>
<td>cevSensorASA5515ChassisTemp (cevSensor 98)</td>
</tr>
<tr>
<td>Central Processing Unit Temperature Sensor for Cisco Adaptive Security Appliance 5515</td>
<td>cevSensorASA5515CPUTemp (cevSensor 97)</td>
</tr>
<tr>
<td>Cisco Adaptive Security Appliance (ASA) 5515 with No Payload Encryption Chassis Fan sensor</td>
<td>cevSensorASA5515K7ChassisFanSensor (cevSensor 126)</td>
</tr>
<tr>
<td>Description</td>
<td>SNMP Object</td>
</tr>
<tr>
<td>-----------------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Central Processing Unit Temperature Sensor for Cisco Adaptive Security Appliance 5515 with No Payload Encryption</td>
<td>cevSensorASA5515K7CPUTemp (cevSensor 103)</td>
</tr>
<tr>
<td>Sensor for Chassis Cooling Fan in Adapative Security Appliance 5515 with No Payload Encryption</td>
<td>cevSensorASA5515K7PSFanSensor (cevSensor 115)</td>
</tr>
<tr>
<td>Sensor for Chassis Cooling Fan in Adapative Security Appliance 5515</td>
<td>cevSensorASA5515PSFanSensor (cevSensor 118)</td>
</tr>
<tr>
<td>Cisco Adaptive Security Appliance (ASA) 5525 Chassis Fan sensor</td>
<td>cevSensorASA5525ChassisFanSensor (cevSensor 122)</td>
</tr>
<tr>
<td>Chassis Ambient Temperature Sensor for Cisco Adaptive Security Appliance 5525</td>
<td>cevSensorASA5525ChassisTemp (cevSensor 108)</td>
</tr>
<tr>
<td>Central Processing Unit Temperature Sensor for Cisco Adaptive Security Appliance 5525</td>
<td>cevSensorASA5525CPUTemp (cevSensor 99)</td>
</tr>
<tr>
<td>Cisco Adaptive Security Appliance (ASA) 5525 with No Payload Encryption Chassis Fan sensor</td>
<td>cevSensorASA5525K7ChassisFanSensor (cevSensor 127)</td>
</tr>
<tr>
<td>Central Processing Unit Temperature Sensor for Cisco Adaptive Security Appliance 5525 with No Payload Encryption</td>
<td>cevSensorASA5525K7CPUTemp (cevSensor 104)</td>
</tr>
<tr>
<td>Sensor for Chassis Cooling Fan in Adapative Security Appliance 5525 with No Payload Encryption</td>
<td>cevSensorASA5525K7PSFanSensor (cevSensor 114)</td>
</tr>
<tr>
<td>Sensor for Chassis Cooling Fan in Adapative Security Appliance 5525</td>
<td>cevSensorASA5525PSFanSensor (cevSensor 117)</td>
</tr>
<tr>
<td>Cisco Adaptive Security Appliance (ASA) 5545 Chassis Fan sensor</td>
<td>cevSensorASA5545ChassisFanSensor (cevSensor 123)</td>
</tr>
<tr>
<td>Chassis Ambient Temperature Sensor for Cisco Adaptive Security Appliance 5545</td>
<td>cevSensorASA5545ChassisTemp (cevSensor 109)</td>
</tr>
<tr>
<td>Central Processing Unit Temperature Sensor for Cisco Adaptive Security Appliance 5545</td>
<td>cevSensorASA5545CPUTemp (cevSensor 100)</td>
</tr>
<tr>
<td>Cisco Adaptive Security Appliance (ASA) 5545 with No Payload Encryption Chassis Fan sensor</td>
<td>cevSensorASA5545K7ChassisFanSensor (cevSensor 128)</td>
</tr>
<tr>
<td>Chassis Ambient Temperature Sensor for Cisco Adaptive Security Appliance 5545 with No Payload Encryption</td>
<td>cevSensorASA5545K7ChassisTemp (cevSensor 90)</td>
</tr>
<tr>
<td>Central Processing Unit Temperature Sensor for Cisco Adaptive Security Appliance 5545 with No Payload Encryption</td>
<td>cevSensorASA5545K7CPUTemp (cevSensor 105)</td>
</tr>
<tr>
<td>Sensor for Chassis Cooling Fan in Adapative Security Appliance 5545 with No Payload Encryption</td>
<td>cevSensorASA5545K7PSFanSensor (cevSensor 113)</td>
</tr>
<tr>
<td>Presence Sensor for Power Supply input in Adapative Security Appliance 5545 with No Payload Encryption</td>
<td>cevSensorASA5545K7PSPresence (cevSensor 87)</td>
</tr>
<tr>
<td>Temperature Sensor for Power Supply Fan in Adapative Security Appliance 5545 with No Payload Encryption</td>
<td>cevSensorASA5545K7PSTempSensor (cevSensor 94)</td>
</tr>
</tbody>
</table>
### Table 54-3  SNMP Physical Vendor Type Values (continued)

<table>
<thead>
<tr>
<th>Sensor Type</th>
<th>Vendor Type Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sensor for Power Supply Fan in Adapative Security Appliance 5545 with No Payload Encryption</td>
<td>cevSensorASA5545PSFanSensor (cevSensor 89)</td>
</tr>
<tr>
<td>Presence Sensor for Power Supply input in Adaptive Security Appliance 5545</td>
<td>cevSensorASA5545PSPresence (cevSensor 130)</td>
</tr>
<tr>
<td>Presence Sensor for Power Supply input in Adaptive Security Appliance 5555</td>
<td>cevSensorASA5545PSPresence (cevSensor 131)</td>
</tr>
<tr>
<td>Temperature Sensor for Power Supply Fan in Adaptive Security Appliance 5545</td>
<td>cevSensorASA5545PSTempSensor (cevSensor 92)</td>
</tr>
<tr>
<td>Cisco Adaptive Security Appliance (ASA) 5555 Chassis Fan sensor</td>
<td>cevSensorASA5555ChassisFanSensor (cevSensor 124)</td>
</tr>
<tr>
<td>Chassis Ambient Temperature Sensor for Cisco Adaptive Security Appliance 5555</td>
<td>cevSensorASA5555ChassisTemp (cevSensor 110)</td>
</tr>
<tr>
<td>Central Processing Unit Temperature Sensor for Cisco Adaptive Security Appliance 5555</td>
<td>cevSensorASA5555CPUTemp (cevSensor 101)</td>
</tr>
<tr>
<td>Cisco Adaptive Security Appliance (ASA) 5555 with No Payload Encryption Chassis Fan sensor</td>
<td>cevSensorASA5555K7ChassisFanSensor (cevSensor 129)</td>
</tr>
<tr>
<td>Chassis Ambient Temperature Sensor for Cisco Adaptive Security Appliance 5555 with No Payload Encryption</td>
<td>cevSensorASA5555K7ChassisTemp (cevSensor 111)</td>
</tr>
<tr>
<td>Central Processing Unit Temperature Sensor for Cisco Adaptive Security Appliance 5555 with No Payload Encryption</td>
<td>cevSensorASA5555K7CPUTemp (cevSensor 106)</td>
</tr>
<tr>
<td>Sensor for Chassis Cooling Fan in Adapative Security Appliance 5555 with No Payload Encryption</td>
<td>cevSensorASA5555K7PSFanSensor (cevSensor 112)</td>
</tr>
<tr>
<td>Presence Sensor for Power Supply input in Adaptive Security Appliance 5555 with No Payload Encryption</td>
<td>cevSensorASA5555K7PSPresence (cevSensor 88)</td>
</tr>
<tr>
<td>Temperature Sensor for Power Supply Fan in Adaptive Security Appliance 5555 with No Payload Encryption</td>
<td>cevSensorASA5555K7PSTempSensor (cevSensor 95)</td>
</tr>
<tr>
<td>Sensor for Power Supply Fan in Adapative Security Appliance 5555</td>
<td>cevSensorASA5555PSFanSensor (cevSensor 91)</td>
</tr>
<tr>
<td>Temperature Sensor for Power Supply Fan in Adapative Security Appliance 5555</td>
<td>cevSensorASA5555PSTempSensor (cevSensor 93)</td>
</tr>
<tr>
<td>Sensor type for ASA 5580</td>
<td>cevSensorASA5580FanSensor (cevSensor 76)</td>
</tr>
<tr>
<td>Sensor for power supply input for ASA 5580</td>
<td>cevSensorASA5580PSInput (cevSensor 74)</td>
</tr>
<tr>
<td>Sensor for power supply fan for ASA 5585-X</td>
<td>cevSensorASA5585PSFanSensor (cevSensor 86)</td>
</tr>
<tr>
<td>Sensor for power supply input for ASA 5585-X</td>
<td>cevSensorASA5585PSInput (cevSensor 85)</td>
</tr>
<tr>
<td>CPU temperature sensor for ASA 5585 SSP-10</td>
<td>cevSensorASA5585SSp10CPUTemp (cevSensor 77)</td>
</tr>
<tr>
<td>CPU temperature sensor for ASA 5585 SSP-10 No Payload Encryption</td>
<td>cevSensorASA5585SSp10K7CPUTemp (cevSensor 78)</td>
</tr>
<tr>
<td>CPU temperature sensor for ASA 5585 SSP-20</td>
<td>cevSensorASA5585SSp20CPUTemp (cevSensor 79)</td>
</tr>
<tr>
<td>CPU temperature sensor for ASA 5585 SSP-20 No Payload Encryption</td>
<td>cevSensorASA5585SSp20K7CPUTemp (cevSensor 80)</td>
</tr>
</tbody>
</table>
## Supported Tables in MIBs

Table 54-4 lists the supported tables and objects for the specified MIBs.

### Table 54-4  Supported Tables and Objects in MIBs

<table>
<thead>
<tr>
<th>MIB Name</th>
<th>Supported Tables and Objects</th>
</tr>
</thead>
<tbody>
<tr>
<td>CISCO-ENHANCED-MEMPOOL-MIB</td>
<td>cempMemPoolTable, cempMemPoolIndex, cempMemPoolType, cempMemPoolName, cempMemPoolAlternate, cempMemPoolValid, cempMemPoolUsed, cempMemPoolFree, cempMemPoolUsedOvrflw, cempMemPoolHCFree</td>
</tr>
<tr>
<td>CISCO-ENTITY-SENSOR-EXT-MIB</td>
<td>Note: Not supported on the ASA Services Module. ceSensorExtThresholdTable</td>
</tr>
<tr>
<td>CISCO-L4L7MODULE-RESOURCE-LIMIT-MIB</td>
<td>CiscoL4L7ResourceLimitTable</td>
</tr>
<tr>
<td>DISMAN-EVENT-MIB</td>
<td>Note: Not supported on the ASA Services Module. mteTriggerTable, mteTriggerThresholdTable, mteObjectsTable, mteEventTable, mteEventNotificationTable</td>
</tr>
<tr>
<td>DISMAN-EXPRESSION-MIB</td>
<td>Note: Not supported on the ASA Services Module. expExpressionTable, expObjectTable, expValueTable</td>
</tr>
<tr>
<td>ENTITY-SENSOR-MIB</td>
<td>Note: Not supported on the ASA Services Module. entPhySensorTable</td>
</tr>
<tr>
<td>NAT-MIB</td>
<td>natAddrMapTable, natAddrMapIndex, natAddrMapName, natAddrMapGlobalAddrType, natAddrMapGlobalAddrFrom, natAddrMapGlobalAddrTo, natAddrMapGlobalPortFrom, natAddrMapGlobalPortTo, natAddrMapProtocol, natAddrMapAddrUsed, natAddrMapRowStatus, cnatAddrBindNumberOfEntries, cnatAddrBindSessionCount</td>
</tr>
</tbody>
</table>
## Supported Traps (Notifications)

Table 54-5 lists the supported traps (notifications) and their associated MIBs.

<table>
<thead>
<tr>
<th>Trap and MIB Name</th>
<th>Varbind List</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>authenticationFailure (SNMPv2-MIB)</td>
<td>—</td>
<td>For SNMP Version 1 or 2, the community string provided in the SNMP request is incorrect. For SNMP Version 3, a report PDU is generated instead of a trap if the auth or priv passwords or usernames are incorrect. The <code>snmp-server enable traps snmp authentication</code> command is used to enable and disable transmission of these traps.</td>
</tr>
<tr>
<td>cefcFRUInserted (CISCO-ENTITY-FRU-CONTROL-MIB)</td>
<td>—</td>
<td>The <code>snmp-server enable traps entity fru-insert</code> command is used to enable this notification.</td>
</tr>
<tr>
<td>cefcFRURemoved (CISCO-ENTITY-FRU-CONTROL-MIB)</td>
<td>—</td>
<td>The <code>snmp-server enable traps entity fru-remove</code> command is used to enable this notification.</td>
</tr>
<tr>
<td>ceSensorExtThresholdNotification (CISCO-ENTITY-SENSOR-EXT-MIB)</td>
<td>ceSensorExtThresholdValue, entPhySensorValue, entPhySensorType, entPhysicalName</td>
<td>The `snmp-server enable traps entity [power-supply-failure</td>
</tr>
<tr>
<td>cipSecTunnelStart (CISCO-IPSEC-FLOW-MONITOR-MIB)</td>
<td>cipSecTunLifeTime, cipSecTunLifeSize</td>
<td>The <code>snmp-server enable traps ipsec start</code> command is used to enable transmission of this trap.</td>
</tr>
<tr>
<td>cipSecTunnelStop (CISCO-IPSEC-FLOW-MONITOR-MIB)</td>
<td>cipSecTunActiveTime</td>
<td>The <code>snmp-server enable traps ipsec stop</code> command is used to enable transmission of this trap.</td>
</tr>
</tbody>
</table>

*Note* Not supported on the ASA Services Module.
### Table 54-5  Supported Traps (Notifications) (continued)

<table>
<thead>
<tr>
<th>Trap Name</th>
<th>Objects</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ciscoRasTooManySessions (CISCO-REMOTE-ACCESS-MONITOR-MIB)</td>
<td>crasNumSessions, crasNumUsers, crasMaxSessionsSupportable, crasMaxUsersSupportable, crasThrMaxSessions</td>
<td>The <code>snmp-server enable traps remote-access session-threshold-exceeded</code> command is used to enable transmission of these traps.</td>
</tr>
<tr>
<td>clogMessageGenerated (CISCO-SYSLOG-MIB)</td>
<td>clogHistFacility, clogHistSeverity, clogHistMsgName, clogHistMsgText, clogHistTimestamp</td>
<td>Syslog messages are generated. The value of the clogMaxSeverity object is used to decide which syslog messages are sent as traps. The <code>snmp-server enable traps syslog</code> command is used to enable and disable transmission of these traps.</td>
</tr>
<tr>
<td>clrResourceLimitReached (CISCO-L4L7MODULE-RESOURCE-LIMIT-MIB)</td>
<td>clrResourceLimitValueType, clrResourceLimitMax, clogOriginIDType, clogOriginID</td>
<td>The <code>snmp-server enable traps connection-limit-reached</code> command is used to enable transmission of the connection-limit-reached notification. The clogOriginID object includes the context name from which the trap originated.</td>
</tr>
<tr>
<td>coldStart (SNMPv2-MIB)</td>
<td>—</td>
<td>The SNMP agent has started. The <code>snmp-server enable traps snmp coldstart</code> command is used to enable and disable transmission of these traps.</td>
</tr>
<tr>
<td>cpmCPURisingThreshold (CISCO-PROCESS-MIB)</td>
<td>cpmCPUrisingThresholdValue, cpmCPUTotalMonIntervalValue, cpmCPUInterruptMonIntervalValue, cpmCPUrisingThresholdPeriod, cpmProcessTimeCreated, cpmProcExtUtil5SecRev</td>
<td>The <code>snmp-server enable traps cpu threshold rising</code> command is used to enable transmission of the cpu threshold rising notification. The cpmCPUrisingThresholdPeriod object is sent with the other objects.</td>
</tr>
<tr>
<td>entConfigChange (ENTITY-MIB)</td>
<td>—</td>
<td>The <code>snmp-server enable traps entity config-change fru-insert fru-remove</code> command is used to enable this notification. Note: This notification is only sent in multimode when a security context is created or removed.</td>
</tr>
<tr>
<td>linkDown (IF-MIB)</td>
<td>ifIndex, ifAdminStatus, ifOperStatus</td>
<td>The linkdown trap for interfaces. The <code>snmp-server enable traps snmp linkdown</code> command is used to enable and disable transmission of these traps.</td>
</tr>
<tr>
<td>linkUp (IF-MIB)</td>
<td>ifIndex, ifAdminStatus, ifOperStatus</td>
<td>The linkup trap for interfaces. The <code>snmp-server enable traps snmp linkup</code> command is used to enable and disable transmission of these traps.</td>
</tr>
</tbody>
</table>
Table 54-5  Supported Traps (Notifications) (continued)

<table>
<thead>
<tr>
<th>mteTriggerFired (DISMAN-EVENT-MIB)</th>
<th>mteHotTrigger, mteHotTargetName, mteHotContextName, mteHotOID, mteHotValue, cempMemPoolName, cempMemPoolHCUsed</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The <code>snmp-server enable traps memory-threshold</code> command is used to enable the memory threshold notification. The mteHotOID is set to cempMemPoolHCUsed. The cempMemPoolName and cempMemPoolHCUsed objects are sent with the other objects.</td>
</tr>
<tr>
<td>mteTriggerFired (DISMAN-EVENT-MIB)</td>
<td>mteHotTrigger, mteHotTargetName, mteHotContextName, mteHotOID, mteHotValue, ifHCInOctets, ifHCOutOctets, ifHighSpeed, entPhysicalName</td>
</tr>
<tr>
<td>Note  Not supported on the ASA Services Module.</td>
<td>The <code>snmp-server enable traps interface-threshold</code> command is used to enable the interface threshold notification. The entPhysicalName objects are sent with the other objects.</td>
</tr>
<tr>
<td>natPacketDiscard (NAT-MIB)</td>
<td>ifIndex</td>
</tr>
<tr>
<td></td>
<td>The <code>snmp-server enable traps nat packet-discard</code> command is used to enable the NAT packet discard notification. This notification is rate limited for 5 minutes and is generated when IP packets are discarded by NAT because mapping space is not available. The ifIndex gives the ID of the mapped interface.</td>
</tr>
<tr>
<td>warmStart (SNMPv2-MIB)</td>
<td>—</td>
</tr>
<tr>
<td></td>
<td>The <code>snmp-server enable traps snmp warmstart</code> command is used to enable and disable transmission of these traps.</td>
</tr>
</tbody>
</table>

SNMP Version 3

This section describes SNMP Version 3 and includes the following topics:

- **SNMP Version 3 Overview**, page 54-14
- **Security Models**, page 54-15
- **SNMP Groups**, page 54-15
- **SNMP Users**, page 54-15
- **SNMP Hosts**, page 54-15
- **Implementation Differences Between the ASASM, ASA Services Module, and the Cisco IOS Software**, page 54-15

SNMP Version 3 Overview

SNMP Version 3 provides security enhancements that are not available in SNMP Version 1 or SNMP Version 2c. SNMP Versions 1 and 2c transmit data between the SNMP server and SNMP agent in clear text. SNMP Version 3 adds authentication and privacy options to secure protocol operations. In addition, this version controls access to the SNMP agent and MIB objects through the User-based Security Model.
(USM) and View-based Access Control Model (VACM). The ASASM/ASASM also support the creation of SNMP groups and users, as well as hosts, which is required to enable transport authentication and encryption for secure SNMP communications.

**Security Models**

For configuration purposes, the authentication and privacy options are grouped together into security models. Security models apply to users and groups, which are divided into the following three types:

- **NoAuthPriv**—No Authentication and No Privacy, which means that no security is applied to messages.
- **AuthNoPriv**—Authentication but No Privacy, which means that messages are authenticated.
- **AuthPriv**—Authentication and Privacy, which means that messages are authenticated and encrypted.

**SNMP Groups**

An SNMP group is an access control policy to which users can be added. Each SNMP group is configured with a security model, and is associated with an SNMP view. A user within an SNMP group must match the security model of the SNMP group. These parameters specify what type of authentication and privacy a user within an SNMP group uses. Each SNMP group name and security model pair must be unique.

**SNMP Users**

SNMP users have a specified username, a group to which the user belongs, authentication password, encryption password, and authentication and encryption algorithms to use. The authentication algorithm options are MD5 and SHA. The encryption algorithm options are DES, 3DES, and AES (which is available in 128, 192, and 256 versions). When you create a user, you must associate it with an SNMP group. The user then inherits the security model of the group.

**SNMP Hosts**

An SNMP host is an IP address to which SNMP notifications and traps are sent. To configure SNMP Version 3 hosts, along with the target IP address, you must configure a username, because traps are only sent to a configured user. SNMP target IP addresses and target parameter names must be unique on the ASASM/ASASM. Each SNMP host can have only one username associated with it. To receive SNMP traps, after you have added the `snmp-server host` command, make sure that you configure the user credentials on the NMS to match the credentials for the ASASM/ASASM.

**Implementation Differences Between the ASASM, ASA Services Module, and the Cisco IOS Software**

The SNMP Version 3 implementation in the ASASM and ASASM differs from the SNMP Version 3 implementation in the Cisco IOS software in the following ways:

- The local-engine and remote-engine IDs are not configurable. The local engine ID is generated when the ASASM/ASASM starts or when a context is created.
- No support exists for view-based access control, which results in unrestricted MIB browsing.
- Support is restricted to the following MIBs: USM, VACM, FRAMEWORK, and TARGET.
Licensing Requirements for SNMP

The following table shows the licensing requirements for this feature:

<table>
<thead>
<tr>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base License: Base (DES).</td>
</tr>
<tr>
<td>Optional license: Strong (3DES, AES)</td>
</tr>
</tbody>
</table>

Prerequisites for SNMP

SNMP has the following prerequisite:

You must have Cisco Works for Windows or another SNMP MIB-II compliant browser to receive SNMP traps or browse a MIB.

Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

Context Mode Guidelines
Supported in single and multiple context mode.

Firewall Mode Guidelines
Supported in routed and transparent firewall mode.

Failover Guidelines
- Supported in SNMP Version 3.
- The SNMP client in each ASASM/ASASM shares engine data with its peer. Engine data includes the engineID, engineBoots, and engineTime objects of the SNMP-FRAMEWORK-MIB. Engine data is written as a binary file to flash:/snmp/contextname.

IPv6 Guidelines
Does not support IPv6.

Additional Guidelines
- Does not support view-based access control, but the VACM MIB is available for browsing to determine default view settings.
- The ENTITY-MIB is not available in the non-admin context. Use the IF-MIB instead to perform queries in the non-admin context.
- Does not support SNMP Version 3 for the AIP SSM or AIP SSC.
- Does not support SNMP debugging.
- Does not support retrieval of ARP information.
- Does not support SNMP SET commands.
- When using NET-SNMP Version 5.4.2.1, only supports the encryption algorithm version of AES128. Does not support the encryption algorithm versions of AES256 or AES192.
- Changes to the existing configuration are rejected if the result places the SNMP feature in an inconsistent state.
- For SNMP Version 3, configuration must occur in the following order: group, user, host.
- Before a group is deleted, you must ensure that all users associated with that group are deleted.
- Before a user is deleted, you must ensure that no hosts are configured that are associated with that username.
- If users have been configured to belong to a particular group with a certain security model, and if the security level of that group is changed, you must do the following in this sequence:
  - Remove the users from that group.
  - Change the group security level.
  - Add users that belong to the new group.
- The creation of custom views to restrict user access to a subset of MIB objects is not supported.
- All requests and traps are available in the default Read/Notify View only.
- The connection-limit-reached trap is generated in the admin context. To generate this trap, you must have at least one snmp-server host configured in the user context in which the connection limit has been reached.
- The value returned for ifNumber will be larger than the number of interfaces that you can query through SNMP, because ifNumber includes hidden internal interfaces that are not viewable.
- You cannot query for the chassis temperature for the ASA 5585 SSP-40 (NPE).

**Configuring SNMP**

This section describes how to configure SNMP and includes the following topics:

- Enabling SNMP, page 54-18
- Configuring SNMP Traps, page 54-19
- Configuring a CPU Usage Threshold, page 54-20
- Configuring a Physical Interface Threshold, page 54-20
- Using SNMP Version 1 or 2c, page 54-21
- Using SNMP Version 3, page 54-22
Enabling SNMP

The SNMP agent that runs on the ASASM performs two functions:

- Replies to SNMP requests from NMSs.
- Sends traps (event notifications) to NMSs.

To enable the SNMP agent and identify an NMS that can connect to the SNMP server, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>snmp-server enable</td>
<td>Ensures that the SNMP server on the ASASM/ASASM is enabled. By default, the SNMP server is enabled.</td>
</tr>
</tbody>
</table>

Example:
```
hostname(config)# snmp-server enable
```
Configuring SNMP Traps

To designate which traps that the SNMP agent generates and how they are collected and sent to NMSs, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>hostname(config)# snmp-server enable traps snmp authentication linkup linkdown coldstart warmstart</code></td>
<td></td>
</tr>
</tbody>
</table>

**Example:**

hostname(config)# snmp-server enable traps snmp authentication linkup linkdown coldstart warmstart

**Note** The `interface-threshold` trap is not supported on the ASASM.

**Purpose**

Sends individual traps, sets of traps, or all traps to the NMS. Enables syslog messages to be sent as traps to the NMS. The default configuration has all SNMP standard traps enabled, as shown in the example. To disable these traps, use the `no snmp-server enable traps snmp` command. If you enter this command and do not specify a trap type, the default is the syslog trap. By default, the syslog trap is enabled. The default SNMP traps continue to be enabled with the syslog trap. You need to configure both the `logging history` command and the `snmp-server enable traps syslog` command to generate traps from the syslog MIB. To restore the default enabling of SNMP traps, use the `clear configure snmp-server` command. All other traps are disabled by default.

**Keywords available in the admin context only:**

- `connection-limit-reached`
- `entity`
- `memory-threshold`

Traps generated through the admin context only for physically connected interfaces in the system context:

- `interface-threshold`

All other traps are available in the admin and user contexts in single mode. In multi-mode, the `fan-failure` trap, the `power-supply-failure` trap, and the `cpu-temperature` trap are generated only from the admin context, and not the user contexts.

If the CPU usage is greater than the configured threshold value for the configured monitoring period, the `cpu threshold rising` trap is generated.

When the used system context memory reaches 80 percent of the total system memory, the `memory-threshold` trap is generated from the admin context. For all other user contexts, this trap is generated when the used memory reaches 80 percent of the total system memory in that particular context.

**Note** SNMP does not monitor voltage sensors.
Configuring SNMP

What to Do Next

See the “Configuring a CPU Usage Threshold” section on page 54-20.

Configuring a CPU Usage Threshold

To configure the CPU usage threshold, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>snmp cpu threshold rising threshold_value monitoring_period</td>
<td>Configures the threshold value for a high CPU threshold and the threshold monitoring period. To clear the threshold value and monitoring period of the CPU utilization, use the no form of this command. If the snmp cpu threshold rising command is not configured, the default for the high threshold level is over 70 percent, and the default for the critical threshold level is over 95 percent. The default monitoring period is set to 1 minute. You cannot configure the critical CPU threshold level, which is maintained at a constant 95 percent. Valid threshold values for a high CPU threshold range from 10 to 94 percent. Valid values for the monitoring period range from 1 to 60 minutes.</td>
</tr>
</tbody>
</table>

Example:
hostname(config)# snmp cpu threshold rising 75% 30 minutes

What to Do Next

See the “Configuring a Physical Interface Threshold” section on page 54-20.

Configuring a Physical Interface Threshold

To configure the physical interface threshold, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>snmp interface threshold threshold_value</td>
<td>Configures the threshold value for an SNMP physical interface. To clear the threshold value for an SNMP physical interface, use the no form of this command. The threshold value is defined as a percentage of interface bandwidth utilization. Valid threshold values range from 30 to 99 percent. The default value is 70 percent. The snmp interface threshold command is available only in the admin context.</td>
</tr>
</tbody>
</table>

Example:
hostname(config)# snmp interface threshold 75%

Note Not supported on the ASA Services Module.

| Note | Physical interface usage is monitored in single mode and multimode, and traps for physical interfaces in the system context are sent through the admin context. Only physical interfaces are used to compute threshold usage. |

What to Do Next

Choose one of the following:
- See the “Using SNMP Version 1 or 2c” section on page 54-21.
- See the “Using SNMP Version 3” section on page 54-22.
## Configuring SNMP

### Using SNMP Version 1 or 2c

To configure parameters for SNMP Version 1 or 2c, perform the following steps:

#### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Specifies the recipient of an SNMP notification, indicates the interface from which traps are sent, and identifies the name and IP address of the NMS or SNMP manager that can connect to the ASASM. The <code>trap</code> keyword limits the NMS to receiving traps only. The <code>poll</code> keyword limits the NMS to sending requests (polling) only. By default, SNMP traps are enabled. By default, the UDP port is 162. The community string is a shared secret key between the ASASM/ASASM and the NMS. The key is a case-sensitive value up to 32 alphanumeric characters. Spaces are not permitted. The default community-string is public. The ASASM uses this key to determine whether the incoming SNMP request is valid. For example, you could designate a site with a community string and then configure the ASASM and the management station with the same string. The ASASM/ASASM uses the specified string and does not respond to requests with an invalid community string. For more information about SNMP hosts, see the “SNMP Hosts” section on page 54-15.</td>
</tr>
</tbody>
</table>
| `snmp-server host` `interface)` `hostname` `ip_address` `[trap | poll] [community community-string] [version {1 | 2c username}] [udp-port port]` | Example: hostname(config)# snmp-server host mgmt 10.7.14.90 version 2  
hostname(config)# snmp-server host corp 172.18.154.159 community public |

| **Step 2** | Sets the community string, which is for use only with SNMP Version 1 or 2c. |
| `snmp-server community` `community-string` | Example: hostname(config)# snmp-server community onceuponatime |

| **Step 3** | Sets the SNMP server location or contact information. |
| `snmp-server [contact | location] text` | Example: hostname(config)# snmp-server location building 42  
hostname(config)# snmp-server contact EmployeeA |
What to Do Next

See the “Monitoring SNMP” section on page 54-25.

Using SNMP Version 3

To configure parameters for SNMP Version 3, perform the following steps:

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Specifies a new SNMP group, which is for use only with SNMP Version 3. When a community string is configured, two additional groups with the name that matches the community string are autogenerated: one for the Version 1 security model and one for the Version 2 security model. For more information about security models, see the “Security Models” section on page 54-15. The auth keyword enables packet authentication. The noauth keyword indicates no packet authentication or encryption is being used. The priv keyword enables packet encryption and authentication. No default values exist for the auth or priv keywords.</td>
</tr>
<tr>
<td>`snmp-server group group-name v3 [auth</td>
<td>noauth</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>hostname(config)# snmp-server group testgroup1 v3 auth</code></td>
<td></td>
</tr>
</tbody>
</table>

| **Step 2** | Configures a new user for an SNMP group, which is for use only with SNMP Version 3. The username argument is the name of the user on the host that belongs to the SNMP agent. The group-name argument is the name of the group to which the user belongs. The v3 keyword specifies that the SNMP Version 3 security model should be used and enables the use of the encrypted, priv, and the auth keywords. The encrypted keyword specifies the password in encrypted format. Encrypted passwords must be in hexadecimal format. The auth keyword specifies which authentication level (md5 or sha) should be used. The priv keyword specifies the encryption level. No default values for the auth or priv keywords, or default passwords exist. For the encryption algorithm, you can specify either the des, 3des, or aes keyword. You can also specify which version of the AES encryption algorithm to use: 128, 192, or 256. The auth-password argument specifies the authentication user password. The priv-password argument specifies the encryption user password. |
| `snmp-server user username group-name (v3 [encrypted]) [auth (md5 | sha)] auth-password [priv (des | 3des | aes) [128 | 192 | 256] priv-password` |  |
| **Example:** |  |
| `hostname(config)# snmp-server user testuser1 testgroup1 v3 auth md5 testpassword aes 128 mypassword` |  |

**Note**  If you forget a password, you cannot recover it and you must reconfigure the user. You can specify a plain-text password or a localized digest. The localized digest must match the authentication algorithm selected for the user, which can be either MD5 or SHA. When the user configuration is displayed on the console or is written to a file (for example, the startup-configuration file), the localized authentication and privacy digests are always displayed instead of a plain-text password (see the second example). The minimum length for a password is 1 alphanumeric character; however, we recommend that you use at least 8 alphanumeric characters for security.
**Troubleshooting Tips**

To ensure that the SNMP process that receives incoming packets from the NMS is running, enter the following command:

```
hostname(config)# show process | grep snmp
```

To capture syslog messages from SNMP and have them appear on the ASASM or ASASM console, enter the following commands:

```
hostname(config)# logging list snmp message 212001-212015
hostname(config)# logging console snmp
```

To ensure that the SNMP process is sending and receiving packets, enter the following commands:

```
hostname(config)# clear snmp-server statistics
hostname(config)# show snmp-server statistics
```

**Step 3**

```
hostname(config)# snmp-server host interface
(hostname | ip_address) [trap | poll] [community
community-string] [version {1 | 2c | 3 username}] [udp-port port]
```

**Example:**

```
hostname(config)# snmp-server host mgmt 10.7.14.90 version 3
   testuser1

hostname(config)# snmp-server host mgmt 10.7.26.5 version 3
   testuser2
```

Specifies the recipient of an SNMP notification. Indicates the interface from which traps are sent. Identifies the name and IP address of the NMS or SNMP manager that can connect to the ASASM. The `trap` keyword limits the NMS to receiving traps only. The `poll` keyword limits the NMS to sending requests (polling) only. By default, SNMP traps are enabled. By default, the UDP port is 162. The community string is a shared secret key between the ASASM and the NMS. The key is a case-sensitive value up to 32 alphanumeric characters. Spaces are not permitted. The default community-string is public. The ASASM/ASASM uses this key to determine whether the incoming SNMP request is valid. For example, you could designate a site with a community string and then configure the ASASM/ASASM with the NMS with the same string. The ASASM/ASASM uses the specified string and does not respond to requests with an invalid community string. For more information about SNMP hosts, see the “SNMP Hosts” section on page 54-15.

**Note** When SNMP Version 3 hosts are configured on the ASASM/ASASM, a user must be associated with that host. To receive traps, after you have added the `snmp-server host` command, make sure that you configure the user on the NMS with the same credentials as the credentials configured on the ASASM/ASASM.

**Step 4**

```
hostname(config)# snmp-server [contact | location] text
```

**Example:**

```
hostname(config)# snmp-server location building 42
hostname(config)# snmp-server contact EmployeeA
```

Sets the SNMP server location or contact information.

**What to Do Next**

See the “Monitoring SNMP” section on page 54-25.
Troubleshooting Tips

The output is based on the SNMP group of the SNMPv2-MIB.

To make sure that SNMP packets are going through the ASASM or ASASM and to the SNMP process, enter the following commands:

```bash
hostname(config)# clear asp drop
hostname(config)# show asp drop
```

If the NMS cannot request objects successfully or is not handing incoming traps from the ASASM or ASASM correctly, use a packet capture to isolate the problem, by entering the following commands:

```bash
hostname (config)# access-list snmp permit udp any eq snmptrap any
hostname (config)# access-list snmp permit udp any any eq snmp
hostname (config)# capture snmp type raw-data access-list snmp interface mgmt
hostname (config)# copy /pcap capture:snmp tftp://192.0.2.5/exampledir/snmp.pcap
```

If the ASASM or ASASM is not performing as expected, obtain information about network topology and traffic by doing the following:

- For the NMS configuration, obtain the following information:
  - Number of timeouts
  - Retry count
  - Engine ID caching
  - Username and password used
- Run the following commands:
  - `show block`
  - `show interface`
  - `show process`
  - `show cpu`

If a fatal error occurs, to help in reproducing the error, send a traceback file and the output of the `show tech-support` command to Cisco TAC.

If SNMP traffic is not being allowed through the ASASM or ASASM interfaces, you might also need to permit ICMP traffic from the remote SNMP server using the `icmp permit` command.

For the ASA 5580, differences may appear in the physical interface statistics output and the logical interface statistics output between the `show interface` command and the `show traffic` command.

## Interface Types and Examples

The interface types that produce SNMP traffic statistics include the following:

- **Logical**—Statistics collected by the software driver, which are a subset of physical statistics.
- **Physical**—Statistics collected by the hardware driver. Each physical named interface has a set of logical and physical statistics associated with it. Each physical interface may have more than one VLAN interface associated with it. VLAN interfaces only have logical statistics.

---

**Note**

For a physical interface that has multiple VLAN interfaces associated with it, be aware that SNMP counters for `ifInOctets` and `ifOutOctets` OIDs match the aggregate traffic counters for that physical interface.
• VLAN-only—SNMP uses logical statistics for ifInOctets and ifOutOctets.

The examples in Table 54-6 show the differences in SNMP traffic statistics. Example 1 shows the difference in physical and logical output statistics for the show interface command and the show traffic command. Example 2 shows output statistics for a VLAN-only interface for the show interface command and the show traffic command. The example shows that the statistics are close to the output that appears for the show traffic command.

Table 54-6    SNMP Traffic Statistics for Physical and VLAN Interfaces

<table>
<thead>
<tr>
<th>Example 1</th>
<th>Example 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>hostname# show interface GigabitEthernet3/2</td>
<td>hostname# show interface GigabitEthernet0/0.100</td>
</tr>
<tr>
<td>interface GigabitEthernet3/2</td>
<td>interface GigabitEthernet0/0.100</td>
</tr>
<tr>
<td>description fullt-mgmt</td>
<td>vlan 100</td>
</tr>
<tr>
<td>nameif mgmt</td>
<td>nameif inside</td>
</tr>
<tr>
<td>security-level 10</td>
<td>security-level 100</td>
</tr>
<tr>
<td>ip address 10.7.14.201 255.255.255.0</td>
<td>ip address 10.7.1.101 255.255.255.0 standby 10.7.1.102</td>
</tr>
<tr>
<td>management-only</td>
<td></td>
</tr>
</tbody>
</table>

hostname# show traffic
(Condensed output)

Physical Statistics
GigabitEthernet3/2:
received (in 121.760 secs)
36 packets 3428 bytes
0 pkts/sec 28 bytes/sec

Logical Statistics
mgmt:
received (in 117.780 secs)
36 packets 2780 bytes
0 pkts/sec 23 bytes/sec

The following examples show the SNMP output statistics for the management interface and the physical interface. The ifInOctets value is close to the physical statistics output that appears in the show traffic command output but not to the logical statistics output.

ifIndex of the mgmt interface:

IF_MIB::ifDescr.6 = Adaptive Security Appliance 'mgmt' interface

ifInOctets that corresponds to the physical interface statistics:

IF-MIB::ifInOctets.6 = Counter32:3246

Table 54-6 SNMP Traffic Statistics for Physical and VLAN Interfaces

<table>
<thead>
<tr>
<th>Example 1</th>
<th>Example 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>hostname# show interface GigabitEthernet3/2</td>
<td>hostname# show interface GigabitEthernet0/0.100</td>
</tr>
<tr>
<td>interface GigabitEthernet3/2</td>
<td>interface GigabitEthernet0/0.100</td>
</tr>
<tr>
<td>description fullt-mgmt</td>
<td>vlan 100</td>
</tr>
<tr>
<td>nameif mgmt</td>
<td>nameif inside</td>
</tr>
<tr>
<td>security-level 10</td>
<td>security-level 100</td>
</tr>
<tr>
<td>ip address 10.7.14.201 255.255.255.0</td>
<td>ip address 10.7.1.101 255.255.255.0 standby 10.7.1.102</td>
</tr>
<tr>
<td>management-only</td>
<td></td>
</tr>
</tbody>
</table>

hostname# show traffic
(Condensed output)

Physical Statistics
GigabitEthernet3/2:
received (in 121.760 secs)
36 packets 3428 bytes
0 pkts/sec 28 bytes/sec

Logical Statistics
mgmt:
received (in 117.780 secs)
36 packets 2780 bytes
0 pkts/sec 23 bytes/sec

The following examples show the SNMP output statistics for the management interface and the physical interface. The ifInOctets value is close to the physical statistics output that appears in the show traffic command output but not to the logical statistics output.

ifIndex of the mgmt interface:

IF_MIB::ifDescr.6 = Adaptive Security Appliance 'mgmt' interface

ifInOctets that corresponds to the physical interface statistics:

IF-MIB::ifInOctets.6 = Counter32:3246

**Monitoring SNMP**

NMSs are the PCs or workstations that you set up to monitor SNMP events and manage devices, such as the ASASM. You can monitor the health of a device from an NMS by polling required information from the SNMP agent that has been set up on the device. Predefined events from the SNMP agent to the NMS generate syslog messages. This section includes the following topics:

- SNMP Syslog Messaging, page 54-26
- SNMP Monitoring, page 54-26
SNMP Syslog Messaging

SNMP generates detailed syslog messages that are numbered 212nnn. Syslog messages indicate the status of SNMP requests, SNMP traps, SNMP channels, and SNMP responses from the ASASM/ASASM to a specified host on a specified interface.

For detailed information about syslog messages, see syslog messages guide.

Note
SNMP polling fails if SNMP syslog messages exceed a high rate (approximately 4000 per second).

SNMP Monitoring

To monitor SNMP, enter one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show running-config snmp-server</td>
<td>Shows all SNMP server configuration information.</td>
</tr>
<tr>
<td>show running-config snmp-server group</td>
<td>Shows SNMP group configuration settings.</td>
</tr>
<tr>
<td>show running-config snmp-server host</td>
<td>Shows configuration settings used by SNMP to control messages and notifications sent to remote hosts.</td>
</tr>
<tr>
<td>show running-config snmp-server user</td>
<td>Shows SNMP user-based configuration settings.</td>
</tr>
<tr>
<td>show snmp-server engineid</td>
<td>Shows the ID of the SNMP engine configured.</td>
</tr>
<tr>
<td>show snmp-server group</td>
<td>Shows the names of configured SNMP groups.</td>
</tr>
<tr>
<td>show snmp-server statistics</td>
<td>Shows the configured characteristics of the SNMP server.</td>
</tr>
<tr>
<td>show snmp-server user</td>
<td>Shows the configured characteristics of users.</td>
</tr>
</tbody>
</table>

Note
If the community string has already been configured, two extra groups appear by default in the output. This behavior is normal.

Examples

The following example shows how to display SNMP server statistics:

hostname(config)# show snmp-server statistics
0 SNMP packets input
  0 Bad SNMP version errors
  0 Unknown community name
  0 Illegal operation for community name supplied
  0 Encoding errors
  0 Number of requested variables
  0 Number of altered variables
  0 Get-request PDUs
  0 Get-next PDUs
Configuration Examples for SNMP

This section includes the following topics:

- Configuration Example for SNMP Versions 1 and 2c, page 54-27
- Configuration Example for SNMP Version 3, page 54-27

Configuration Example for SNMP Versions 1 and 2c

The following example shows how the ASASM can receive SNMP requests from host 192.0.2.5 on the inside interface but does not send any SNMP syslog requests to any host:

```
hostname(config)# snmp-server host 192.0.2.5
hostname(config)# snmp-server location building 42
hostname(config)# snmp-server contact EmployeeA
hostname(config)# snmp-server community ohwhatakeyistheh
```

Configuration Example for SNMP Version 3

The following example shows how the ASASM can receive SNMP requests using the SNMP Version 3 security model, which requires that the configuration follow this specific order: group, followed by user, followed by host:

```
hostname(config)# snmp-server group v3 vpn-group priv
hostname(config)# snmp-server user admin vpn group v3 auth sha letmein priv 3des cisco123
hostname(config)# snmp-server host mgmt 10.0.0.1 version 3 priv admin
```
Where to Go Next

To configure the syslog server, see Chapter 52, “Configuring Logging.”

Additional References

For additional information related to implementing SNMP, see the following sections:

- RFCs for SNMP Version 3, page 54-28
- MIBs, page 54-28
- Application Services and Third-Party Tools, page 54-30

RFCs for SNMP Version 3

<table>
<thead>
<tr>
<th>RFC</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>3410</td>
<td>Introduction and Applicability Statements for Internet Standard Management Framework</td>
</tr>
<tr>
<td>3411</td>
<td>An Architecture for Describing SNMP Management Frameworks</td>
</tr>
<tr>
<td>3412</td>
<td>Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)</td>
</tr>
<tr>
<td>3413</td>
<td>Simple Network Management Protocol (SNMP) Applications</td>
</tr>
<tr>
<td>3414</td>
<td>User-based Security Model (USM) for Version 3 of the Simple Network Management Protocol (SNMP)</td>
</tr>
<tr>
<td>3826</td>
<td>The Advanced Encryption Standard (AES) Cipher Algorithm in the SNMP User-based Security Model</td>
</tr>
</tbody>
</table>

MIBs

For a list of supported MIBs and traps for the ASASM/ASASMby release, see the following URL: http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

Not all OIDs in MIBs are supported. To obtain a list of the supported SNMP MIBs and OIDs for a specific ASASM/ASASM, enter the following command:

```
hostname(config)# show snmp-server oidlist
```

Although the **oidlist** keyword does not appear in the options list for the **show snmp-server** command help, it is available. However, this command is for Cisco TAC use only. Contact the Cisco TAC before using this command.

The following is sample output from the **show snmp-server oidlist** command:

```
hostname(config)# show snmp-server oidlist
[0] 1.3.6.1.2.1.1.1.1. sysDescr
[1] 1.3.6.1.2.1.1.2.  sysObjectID
[2] 1.3.6.1.2.1.1.3.  sysUpTime
[3] 1.3.6.1.2.1.1.4.  sysContact
[4] 1.3.6.1.2.1.1.5.  sysName
[5] 1.3.6.1.2.1.1.6.  sysLocation
```
<table>
<thead>
<tr>
<th>No.</th>
<th>OID</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>1.3.6.1.2.1.1.7.</td>
<td>sysServices</td>
</tr>
<tr>
<td>7</td>
<td>1.3.6.1.2.1.2.1.</td>
<td>ifNumber</td>
</tr>
<tr>
<td>8</td>
<td>1.3.6.1.2.1.2.2.1.</td>
<td>ifIndex</td>
</tr>
<tr>
<td>9</td>
<td>1.3.6.1.2.1.2.2.1.2.</td>
<td>ifDescr</td>
</tr>
<tr>
<td>10</td>
<td>1.3.6.1.2.1.2.2.1.3.</td>
<td>ifType</td>
</tr>
<tr>
<td>11</td>
<td>1.3.6.1.2.1.2.2.1.4.</td>
<td>ifMtu</td>
</tr>
<tr>
<td>12</td>
<td>1.3.6.1.2.1.2.2.1.5.</td>
<td>ifSpeed</td>
</tr>
<tr>
<td>13</td>
<td>1.3.6.1.2.1.2.2.1.6.</td>
<td>ifPhysAddress</td>
</tr>
<tr>
<td>14</td>
<td>1.3.6.1.2.1.2.2.1.7.</td>
<td>ifAdminStatus</td>
</tr>
<tr>
<td>15</td>
<td>1.3.6.1.2.1.2.2.1.8.</td>
<td>ifOperStatus</td>
</tr>
<tr>
<td>16</td>
<td>1.3.6.1.2.1.2.2.1.9.</td>
<td>ifLastChange</td>
</tr>
<tr>
<td>17</td>
<td>1.3.6.1.2.1.2.2.2.1.10.</td>
<td>ifInOctets</td>
</tr>
<tr>
<td>18</td>
<td>1.3.6.1.2.1.2.2.2.1.11.</td>
<td>ifInUCastPkts</td>
</tr>
<tr>
<td>19</td>
<td>1.3.6.1.2.1.2.2.2.1.12.</td>
<td>ifInNUcastPkts</td>
</tr>
<tr>
<td>20</td>
<td>1.3.6.1.2.1.2.2.2.1.13.</td>
<td>ifInDiscards</td>
</tr>
<tr>
<td>21</td>
<td>1.3.6.1.2.1.2.2.2.1.14.</td>
<td>ifInErrors</td>
</tr>
<tr>
<td>22</td>
<td>1.3.6.1.2.1.2.2.2.1.15.</td>
<td>ifOutOctets</td>
</tr>
<tr>
<td>23</td>
<td>1.3.6.1.2.1.2.2.2.1.16.</td>
<td>ifOutUCastPkts</td>
</tr>
<tr>
<td>24</td>
<td>1.3.6.1.2.1.2.2.2.1.17.</td>
<td>ifOutNUcastPkts</td>
</tr>
<tr>
<td>25</td>
<td>1.3.6.1.2.1.2.2.2.1.18.</td>
<td>ifOutDiscards</td>
</tr>
<tr>
<td>26</td>
<td>1.3.6.1.2.1.2.2.2.1.19.</td>
<td>ifOutErrors</td>
</tr>
<tr>
<td>27</td>
<td>1.3.6.1.2.1.2.2.2.1.20.</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>1.3.6.1.2.1.2.2.2.1.21.</td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>1.3.6.1.2.1.2.2.2.1.22.</td>
<td>ifSpecific</td>
</tr>
<tr>
<td>30</td>
<td>1.3.6.1.2.1.4.1.</td>
<td>ipForwarding</td>
</tr>
<tr>
<td>31</td>
<td>1.3.6.1.2.1.4.20.1.1.</td>
<td>ipAdEntAddr</td>
</tr>
<tr>
<td>32</td>
<td>1.3.6.1.2.1.4.20.1.2.</td>
<td>ipAdEntIfIndex</td>
</tr>
<tr>
<td>33</td>
<td>1.3.6.1.2.1.4.20.1.3.</td>
<td>ipAdEntNetMask</td>
</tr>
<tr>
<td>34</td>
<td>1.3.6.1.2.1.4.20.1.4.</td>
<td>ipAdEntBcastAddr</td>
</tr>
<tr>
<td>35</td>
<td>1.3.6.1.2.1.1.1.11.</td>
<td>snmpInPkts</td>
</tr>
<tr>
<td>36</td>
<td>1.3.6.1.2.1.1.11.2.</td>
<td>snmpOutPkts</td>
</tr>
<tr>
<td>37</td>
<td>1.3.6.1.2.1.1.11.3.</td>
<td>snmpInBadVersions</td>
</tr>
<tr>
<td>38</td>
<td>1.3.6.1.2.1.1.11.4.</td>
<td>snmpInBadCommunityNames</td>
</tr>
<tr>
<td>39</td>
<td>1.3.6.1.2.1.1.11.5.</td>
<td>snmpInBadCommunityUses</td>
</tr>
<tr>
<td>40</td>
<td>1.3.6.1.2.1.1.11.6.</td>
<td>snmpInASNParseErrs</td>
</tr>
<tr>
<td>41</td>
<td>1.3.6.1.2.1.1.11.8.</td>
<td>snmpInTooBigs</td>
</tr>
<tr>
<td>42</td>
<td>1.3.6.1.2.1.1.11.9.</td>
<td>snmpInNoSuchNames</td>
</tr>
<tr>
<td>43</td>
<td>1.3.6.1.2.1.1.11.10.</td>
<td>snmpInBadValues</td>
</tr>
<tr>
<td>44</td>
<td>1.3.6.1.2.1.1.11.11.</td>
<td>snmpInReadOnlys</td>
</tr>
<tr>
<td>45</td>
<td>1.3.6.1.2.1.1.11.12.</td>
<td>snmpInGenErrs</td>
</tr>
<tr>
<td>46</td>
<td>1.3.6.1.2.1.1.11.13.</td>
<td>snmpInTotalReqVars</td>
</tr>
<tr>
<td>47</td>
<td>1.3.6.1.2.1.1.11.14.</td>
<td>snmpInTotalSetVars</td>
</tr>
<tr>
<td>48</td>
<td>1.3.6.1.2.1.1.11.15.</td>
<td>snmpInGetRequests</td>
</tr>
<tr>
<td>49</td>
<td>1.3.6.1.2.1.1.11.16.</td>
<td>snmpInGetNexts</td>
</tr>
<tr>
<td>50</td>
<td>1.3.6.1.2.1.1.11.17.</td>
<td>snmpInGetRequests</td>
</tr>
<tr>
<td>51</td>
<td>1.3.6.1.2.1.1.11.18.</td>
<td>snmpInGetResponses</td>
</tr>
<tr>
<td>52</td>
<td>1.3.6.1.2.1.1.11.19.</td>
<td>snmpInTraps</td>
</tr>
<tr>
<td>53</td>
<td>1.3.6.1.2.1.1.11.20.</td>
<td>snmpOutTooBigs</td>
</tr>
<tr>
<td>54</td>
<td>1.3.6.1.2.1.1.11.21.</td>
<td>snmpOutNoSuchNames</td>
</tr>
<tr>
<td>55</td>
<td>1.3.6.1.2.1.1.11.22.</td>
<td>snmpOutBadValues</td>
</tr>
<tr>
<td>56</td>
<td>1.3.6.1.2.1.1.11.24.</td>
<td>snmpOutGenErrs</td>
</tr>
<tr>
<td>57</td>
<td>1.3.6.1.2.1.1.11.25.</td>
<td>snmpOutGetRequests</td>
</tr>
<tr>
<td>58</td>
<td>1.3.6.1.2.1.1.11.26.</td>
<td>snmpOutGetNexts</td>
</tr>
<tr>
<td>59</td>
<td>1.3.6.1.2.1.1.11.27.</td>
<td>snmpOutGetRequests</td>
</tr>
<tr>
<td>60</td>
<td>1.3.6.1.2.1.1.11.28.</td>
<td>snmpOutGetResponses</td>
</tr>
<tr>
<td>61</td>
<td>1.3.6.1.2.1.1.11.29.</td>
<td>snmpOutTraps</td>
</tr>
<tr>
<td>62</td>
<td>1.3.6.1.2.1.1.11.30.</td>
<td>snmpEnableAuthenTraps</td>
</tr>
<tr>
<td>63</td>
<td>1.3.6.1.2.1.1.11.31.</td>
<td>snmpSilentDrops</td>
</tr>
<tr>
<td>64</td>
<td>1.3.6.1.2.1.1.11.32.</td>
<td>snmpProxyDrops</td>
</tr>
<tr>
<td>65</td>
<td>1.3.6.1.2.1.31.1.1.1.1.</td>
<td>ifName</td>
</tr>
<tr>
<td>66</td>
<td>1.3.6.1.2.1.31.1.1.1.2.</td>
<td>inMulticastPkts</td>
</tr>
<tr>
<td>67</td>
<td>1.3.6.1.2.1.31.1.1.1.3.</td>
<td>inBroadcastPkts</td>
</tr>
<tr>
<td>68</td>
<td>1.3.6.1.2.1.31.1.1.1.4.</td>
<td>outMulticastPkts</td>
</tr>
<tr>
<td>69</td>
<td>1.3.6.1.2.1.31.1.1.1.5.</td>
<td>outBroadcastPkts</td>
</tr>
</tbody>
</table>
Feature History for SNMP

Table 54-7  Feature History for SNMP

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>SNMP Versions 1</td>
<td>7.0(1)</td>
<td>Provides ASASM/ASASM network monitoring and event information by transmitting data between the SNMP server and SNMP agent through the clear text community string.</td>
</tr>
<tr>
<td>and 2c</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SNMP Version 3</td>
<td>8.2(1)</td>
<td>Provides 3DES or AES encryption and support for SNMP Version 3, the most secure form of the supported security models. This version allows you to configure users, groups, and hosts, as well as authentication characteristics by using the USM. In addition, this version allows access control to the agent and MIB objects and includes additional MIB support.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>We introduced or modified the following commands: <strong>show snmp-server engineid</strong>, <strong>show snmp-server group</strong>, <strong>show snmp-server user</strong>, <strong>snmp-server user</strong>, <strong>snmp-server host</strong>.</td>
</tr>
<tr>
<td>Password encryption</td>
<td>8.3(1)</td>
<td>Supports password encryption.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>We modified the following commands: <strong>snmp-server community</strong>, <strong>snmp-server host</strong>.</td>
</tr>
</tbody>
</table>
### Table 54-7 Feature History for SNMP (continued)

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
</table>
| SNMP traps and MIBs | 8.4(1) | Supports the following additional keywords: connection-limit-reached, cpu threshold rising, entity cpu-temperature, entity fan-failure, entity power-supply, ikev2 stop | start, interface-threshold, memory-threshold, nat packet discard, warmstart.  
The entPhysicalTable reports entries for sensors, fans, power supplies, and related components.  
Supports the following additional MIBs: CISCO-ENTITY-SENSOR-EXT-MIB, CISCO-ENTITY-FRU-CONTROL-MIB, CISCO-PROCESS-MIB, CISCO-ENHANCED-MEMPOOL-MIB, CISCO-L4L7MODULE-RESOURCE-LIMIT-MIB, DISMAN-EVENT-MIB, DISMAN-EXPRESSION-MIB, ENTITY-SENSOR-MIB, NAT-MIB.  
Supports the following additional traps: ceSensorExtThresholdNotification, clrResourceLimitReached, cpmCPUrisingThreshold, mteTriggerFired, natPacketDiscard, warmStart.  
We introduced or modified the following commands: snmp cpu threshold rising, snmp interface threshold, snmp-server enable traps. |
| IF-MIB ifAlias OID support | 8.2(5)/8.4(2) | The ASASM now supports the ifAlias OID. When you browse the IF-MIB, the ifAlias OID will be set to the value that has been set for the interface description. |
| ASA Services Module (ASASM) | 8.5(1) | The ASASM supports all MIBs and traps that are present in 8.4(1), except for the following:  
Unsupported MIBs in 8.5(1):  
• CISCO-ENTITY-SENSOR-EXT-MIB (Only objects under the entPhySensorTable group are supported).  
• ENTITY-SENSOR-MIB (Only objects in the entPhySensorTable group are supported).  
• DISMAN-EXPRESSION-MIB (Only objects in the expExpressionTable, expObjectTable, and expValueTable groups are supported).  
Unsupported traps in 8.5(1):  
• ceSensorExtThresholdNotification (CISCO-ENTITY-SENSOR-EXT-MIB).  
  This trap is only used for power supply failure, fan failure, and high CPU temperature events.  
• InterfacesBandwidthUtilization. |
| SNMP traps | 8.6(1) | Supports the following additional keywords for the ASA 5512-X, 5515-X, 5525-X, 5545-X, and 5555-X: entity power-supply-presence, entity power-supply-failure, entity chassis-temperature, entity chassis-fan Failure, entity power-supply-temperature.  
We modified the following command: snmp-server enable traps. |
| NAT MIB | 8.4(5) | Added the cnatAddrBindNumberOfEntries and cnatAddrBindSessionCount OIDs to support the xlate_count and max_xlate_count entries, which are the equivalent to allowing polling using the show xlate count command. |
Configuring Anonymous Reporting and Smart Call Home

The Smart Call Home feature provides personalized, e-mail-based and web-based notification to customers about critical events involving their individual systems, often before customers know that a critical event has occurred.

The Anonymous Reporting feature is a subfeature of the Smart Call Home feature and allows Cisco to anonymously receive minimal error and health information from the device.

You might have received a popup dialog that invites you to do the following:

- Enable Anonymous Reporting to help improve the ASA platform.
- Register for Smart Home Notifications to receive personalized, proactive assistance from Cisco.

For information about the dialog, see the “Anonymous Reporting and Smart Call Home Prompt” section on page 55-3.

This chapter describes how to use and configure Anonymous Reporting and Smart Call Home, and it includes the following sections:

- Information About Anonymous Reporting and Smart Call Home, page 55-1
- Licensing Requirements for Anonymous Reporting and Smart Call Home, page 55-4
- Prerequisites for Smart Call Home and Anonymous Reporting, page 55-5
- Guidelines and Limitations, page 55-5
- Configuring Anonymous Reporting and Smart Call Home, page 55-6
- Monitoring Smart Call Home, page 55-19
- Configuration Example for Smart Call Home, page 55-19
- Feature History for Anonymous Reporting and Smart Call Home, page 55-20

Information About Anonymous Reporting and Smart Call Home

This section includes the following topics:

- Information About Anonymous Reporting, page 55-2
- Information About Smart Call Home, page 55-4
Information About Anonymous Reporting

Customers can help to improve the ASA platform by enabling Anonymous Reporting, which allows Cisco to securely receive minimal error and health information from the device. If you enable the feature, your customer identity will remain anonymous, and no identifying information will be sent.

Enabling Anonymous Reporting creates a trust point and installs a certificate. A CA certificate is required for your ASA to validate the server certificate present on the Smart Call Home web server and to form the HTTPS session so that your ASA can send messages securely. Cisco imports a certificate that is predefined in the software. If you decide to enable Anonymous Reporting, a certificate is installed on the ASA with a hardcoded trust point name: _SmartCallHome_ServerCA. When you enable Anonymous Reporting, this trust point is created, the appropriate certificate is installed, and you receive a message about this action. The certificate then shows up in your configuration.

If the appropriate certificate already exists in your configuration when you enable Anonymous Reporting, no trust point is created, and no certificate is installed.

Note
When you enable Anonymous Reporting you acknowledge your consent to transfer the specified data to Cisco or to vendors operating on Cisco’s behalf (including countries outside of the U.S.).
Cisco maintains the privacy of all customers. For information about Cisco’s treatment of personal information, see the Cisco Privacy Statement at the following URL: http://www.cisco.com/web/siteassets/legal/privacy.html

What is Sent to Cisco?

Messages are sent to Cisco once a month and whenever the ASA reloads. These messages are categorized by alert groups, which are predefined subsets of Smart Call Home alerts that are supported on the ASA: configuration alerts, inventory alerts, and crash information alerts.

Inventory alerts consist of output from the following commands:

- **show version**—Displays the ASA software version, hardware configuration, license key, and related uptime data for the device.
- **show environment**—Shows system environment information for ASA system components, such as hardware operational status for the chassis, drivers, fans, and power supplies, as well as temperature status, voltage, and CPU usage.
- **show inventory**—Retrieves and displays inventory information about each Cisco product that is installed in the networking device. Each product is identified by unique device information, called the UDI, which is a combination of three separate data elements: the product identifier (PID), the version identifier (VID), and the serial number (SN).
- **show failover state**—Displays the failover state of both units in a failover pair. The information displayed includes the primary or secondary status of the unit, the Active/Standby status of the unit, and the last reported reason for failover.
- **show module**—Shows information about any modules installed on the ASAs, for example, information about an AIP SSC installed on the ASA 5505 or information about an SSP installed on the ASA 5585-X, and information about an IPS SSP installed on an ASA 5585-X.

Configuration alerts consist of output from the following commands:

- **show context**—Shows allocated interfaces and the configuration file URL, the number of contexts configured, or, if you enable AR in the system execution space, from a list of all contexts.
Information About Anonymous Reporting and Smart Call Home

- **show call-home registered-module status**—Displays the registered module status. If you use system configuration mode, the command displays system module status based on the entire device, not per context.

Upon a system crash, modified information from the following command is sent:

- **show crashinfo** (truncated)—Upon an unexpected software reload, the device sends a modified crash information file with only the traceback section of the file included, so only function calls, register values, and stack dumps are reported to Cisco.

For more information about ASA commands, see the Cisco ASA 5500 Series Command Reference document.

DNS Requirement

A DNS server must be configured properly for your ASA to reach the Cisco Smart Call Home server and send messages to Cisco. Because it is possible that your ASA resides in a private network and does not have access to the public network, Cisco verifies your DNS configuration and then configures it for you, if necessary, by doing the following:

1. Performing a DNS lookup for all DNS servers configured.
2. Getting the DNS server from the DHCP server by sending DHCPINFORM messages on the highest security-level interface.
3. Using the Cisco DNS servers for lookup.

The above tasks are performed without changing the current configuration. (For example, the DNS server learned from DHCP will not be added to the configuration.)

If there is no DNS server configured, and your ASA cannot reach the Cisco Smart Call Home Server, Cisco generates a syslog message with the “warning” severity for every Smart Call Home message sent to remind you to configure DNS properly.

For information about system log messages, see the *Cisco ASA 5500 Series System Log Messages*.

Anonymous Reporting and Smart Call Home Prompt

When you enter configuration mode you receive a prompt that invites you to enable the Anonymous Reporting and Smart Call Home features if the following criteria are met:

At the prompt you may choose [Y]es, [N]o, [A]sk later. If you choose [A]sk later, then you are reminded again in seven days or when the ASA reloads. If you continue to choose [A]sk later, the ASA prompts two more times at seven-day intervals before it assumes a [N]o response and does not ask again.

At the ASDM prompt you can select from the following options:

- Registered (enter an e-mail address)—Enables Smart Call Home and registers your ASA with Cisco TAC.
- Do not enable Smart Call Home—Does not enable Smart Call Home and does not ask again.
- Remind Me Later—Defers the decision. You are reminded again in seven days or whenever the ASA reloads. The ASA prompts two more times at seven-day intervals before it assumes a “Do not enable Smart Call Home response” and does not ask again.
If you did not receive the prompt, you may enable Anonymous Reporting or Smart Call Home by performing the steps in the “Configuring Anonymous Reporting” section on page 55-6 or the “Configuring Smart Call Home” section on page 55-7.

Information About Smart Call Home

When fully configured, Smart Call Home detects issues at your site and reports them back to Cisco or through other user-defined channels (such as e-mail or directly to you), often before you know that these issues exist. Depending upon the seriousness of these problems, Cisco responds to customers regarding their system configuration issues, product end-of-life announcements, security advisory issues, and so on.

In this manner, Smart Call Home offers proactive diagnostics and real-time alerts on the ASASM and provides high network availability and increased operational efficiency through proactive and quick issue resolution by doing the following:

- Identifying issues quickly with continuous monitoring, real-time proactive alerts, and detailed diagnostics.
- Making you aware of potential problems through Smart Call Home notifications, in which a service request has been opened, with all diagnostic data attached.
- Resolving critical problems faster with direct, automatic access to experts in Cisco TAC.

Smart Call Home offers increased operational efficiency by providing you with the ability to do the following:

- Use staff resources more efficiently by reducing troubleshooting time.
- Generate service requests to Cisco TAC automatically, routed to the appropriate support team, which provides detailed diagnostic information that speeds problem resolution.

The Smart Call Home Portal offers quick, web-based access to required information that provides you with the ability to do the following:

- Review all Smart Call Home messages, diagnostics, and recommendations in one place.
- Check service request status quickly.
- View the most up-to-date inventory and configuration information for all Smart Call Home-enabled devices.

Licensing Requirements for Anonymous Reporting and Smart Call Home

The following table shows the licensing requirements for Anonymous Reporting and Smart Call Home:

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>All models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>
Prerequisites for Smart Call Home and Anonymous Reporting

Smart Call Home and Anonymous Reporting have the following prerequisites:

- DNS must be configured. (See the “DNS Requirement” section on page 55-3 and see the “Configuring the DNS Server” section on page 9-8.)

Guidelines and Limitations

Firewall Mode Guidelines
Supported in routed and transparent firewall modes.

Context Mode Guidelines
Supported in single mode and multiple context mode.

IPv6 Guidelines
Supports IPv6.

Additional Guidelines for Anonymous Reporting

- If an Anonymous Reporting message cannot be sent on the first try, the ASA retries two more times before dropping the message.
- Anonymous Reporting can coexist with other Smart Call Home configurations without changing the existing configuration. For example, if Smart Call Home is off before enabling Anonymous Reporting, it remains off, even after enabling Anonymous Reporting.
- Output from the `show running-config all` command shows details about the Anonymous Reporting user profile.
- If Anonymous Reporting is enabled, you cannot remove the trust point, and when Anonymous Reporting is disabled, the trust point remains. If Anonymous Reporting is disabled, users can remove the trustpoint, but disabling Anonymous Reporting will not cause the trustpoint to be removed.

Additional Guidelines for Smart Call Home

- In multiple context mode, the `snapshots` command is divided into two commands: one to obtain information from the system context and one to obtain information from the regular context.
- The Smart Call Home back-end server can accept messages in XML format only.
Configuring Anonymous Reporting and Smart Call Home

While Anonymous Reporting is a subfeature of the Smart Call Home feature and allows Cisco to anonymously receive minimal error and health information from the device, the Smart Call Home feature is more robust and allows for customized support of your system health, allowing Cisco TAC to monitor your devices and open a case when there is an issue, often before you know the issue occurred.

Generally speaking, you can have both features configured on your system at the same time, yet configuring the robust Smart Call Home feature provides the same functionality as Anonymous reporting, plus personalized service.

This section includes the following topics:
- Configuring Anonymous Reporting, page 55-6
- Configuring Smart Call Home, page 55-7

Configuring Anonymous Reporting

To configure Anonymous Reporting and securely provide minimal error and health information to Cisco, perform the following steps:

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> call-home reporting anonymous</td>
<td>Enables the Anonymous Reporting feature and creates a new anonymous profile. Entering this command creates a trust point and installs a certificate that is used to verify the identity of the Cisco web server.</td>
</tr>
<tr>
<td>Example: hostname(config)# call-home reporting anonymous</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> call-home test reporting anonymous</td>
<td>(Optional) Tests that the Anonymous Reporting feature is fully enabled. Also ensures that you have connectivity to the server and that your system is able to send messages. A success or error message returns test results.</td>
</tr>
<tr>
<td>Example: hostname(config)# call-home test reporting anonymous</td>
<td></td>
</tr>
</tbody>
</table>
Configuring Smart Call Home

This section describes how to configure the Smart Call Home feature.

This section includes the following topics:

- Enabling Smart Call Home, page 55-7
- Declaring and Authenticating a CA Trust Point, page 55-8
- Configuring DNS, page 55-8
- Subscribing to Alert Groups, page 55-9
- Testing Call Home Communications, page 55-11
- Optional Configuration Procedures, page 55-13

Enabling Smart Call Home

This section contains information about performing basic setup for the Smart Call Home feature.

To enable Smart Call Home and activate your call-home profile, perform this task:

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>service call-home</td>
<td>Enables the smart call home service.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# service call-home</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>call-home</td>
<td>Enters call-home configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# call-home</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>contact-email-addr email</td>
<td>Configures the mandatory contact address. The address should be the Cisco.com ID account associated with the device.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(cfg-call-home)# contact-email-addr <a href="mailto:username@example.com">username@example.com</a></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>profile profile-name</td>
<td>Enables the profile. The default profile name is CiscoTAC-1.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(cfg-call-home)# profile CiscoTAC-1</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>active</td>
<td>Activates the call home profile. To disable this profile, enter the no active command.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(cfg-call-home-profile)# active</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>destination transport-method http</td>
<td>Configures the destination transport method for the smart call-home message receiver. The default destination transport method is e-mail. To configure e-mail see the “Sending the Output of a Command” section on page 55-12.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(cfg-call-home-profile)# destination transport-method http</td>
<td></td>
</tr>
</tbody>
</table>
Declaring and Authenticating a CA Trust Point

If Smart Call Home is configured to send messages to a web server through HTTPS, you need to configure the ASA to trust the certificate of the web server or the certificate of the Certificate Authority (CA) that issued the certificate. The Cisco Smart Call Home Production server certificate is issued by Verisign. The Cisco Smart Call Home Staging server certificate is issued by Digital Signature Trust Co.

Detailed Steps

To declare and authenticate the Cisco server security certificate and establish communication with the Cisco HTTPS server for Smart Call Home service, perform this task:

<table>
<thead>
<tr>
<th>Step 1</th>
<th>crypto ca trustpoint trustpoint-name</th>
<th>Configures a trustpoint and prepares for certificate enrollment.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Example:</td>
<td>hostname(config)# crypto ca trustpoint cisco</td>
<td>Note: If you use HTTP as the transport method, you must install a security certificate through a trustpoint, which is required for HTTPS. Find the specific certificate to install at the following URL: <a href="http://www.cisco.com/en/US/docs/switches/lan/smart_call_home/SCH31_Ch6.html#wp1035380">http://www.cisco.com/en/US/docs/switches/lan/smart_call_home/SCH31_Ch6.html#wp1035380</a></td>
</tr>
</tbody>
</table>

| Example: | hostname(ca-trustpoint)# enroll terminal |

| Step 3 | exit | Exits CA trustpoint configuration mode and returns to global configuration mode. |
| Example: | hostname(ca-trustpoint)# exit |

| Step 4 | crypto ca authenticate trustpoint | Authenticates the named CA. The CA name should match the trust point name specified in the crypto ca trustpoint command. At the prompt, paste the security certificate text. |
| Example: | hostname(ca-trustpoint)# crypto ca authenticate cisco |

| Step 5 | quit | Specifies the end of the security certificate text and confirms acceptance of the entered security certificate. |
| Example: | hostname(ca-trustpoint)# quit |

%Do you accept this certificate [yes/no]: yes

Configuring DNS

You must configure DNS so that the HTTPS URLs in the Smart Call Home profile can successfully resolve.

To configure DNS, perform the following tasks:
### Configuring Periodic Notification

When you subscribe a destination profile to either the Configuration or the Inventory alert group, you can choose to receive the alert group messages asynchronously or periodically at a specified time. The sending period can be one of the following:

- **Daily**—Specify the time of the day to send, using an hour:minute format `hh:mm`, with a 24-hour clock (for example, 14:30).
- **Weekly**—Specify the day of the week and time of day in the format `day hh:mm`, where the day of the week is spelled out (for example, monday).
- **Monthly**—Specify the numeric date, from 1 to 31, and the time of day, in the format `date hh:mm`.

### Information about the Message Severity Threshold

When you subscribe a destination profile to certain alert groups, you can set a threshold for sending alert group messages based upon the message level severity. (See Table 55-1). Any message with a value lower than the destination profile’s specified threshold is not sent to the destination.
Table 55-1  Severity and Syslog Level Mapping

<table>
<thead>
<tr>
<th>Level</th>
<th>Keyword</th>
<th>Equivalent Syslog Level</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td>catastrophic</td>
<td>N/A</td>
<td>Network-wide catastrophic failure.</td>
</tr>
<tr>
<td>8</td>
<td>disaster</td>
<td>N/A</td>
<td>Significant network impact.</td>
</tr>
<tr>
<td>7</td>
<td>fatal</td>
<td>Emergency (0)</td>
<td>System is unusable.</td>
</tr>
<tr>
<td>6</td>
<td>critical</td>
<td>Alert (1)</td>
<td>Critical conditions, immediate attention needed.</td>
</tr>
<tr>
<td>5</td>
<td>major</td>
<td>Critical (2)</td>
<td>Major conditions.</td>
</tr>
<tr>
<td>4</td>
<td>minor</td>
<td>Error (3)</td>
<td>Minor conditions.</td>
</tr>
<tr>
<td>3</td>
<td>warning</td>
<td>Warning (4)</td>
<td>Warning conditions.</td>
</tr>
<tr>
<td>2</td>
<td>notification</td>
<td>Notice (5)</td>
<td>Basic notification and informational messages. Possibly independently insignificant.</td>
</tr>
<tr>
<td>1</td>
<td>normal</td>
<td>Information (6)</td>
<td>Normal event signifying return to normal state.</td>
</tr>
<tr>
<td>0</td>
<td>debugging</td>
<td>Debug (7)</td>
<td>Debugging messages (default setting).</td>
</tr>
</tbody>
</table>

Configuring Alert Group Subscription

To subscribe a destination profile to an alert group, perform this task:

Detailed Steps

**Step 1**

**Command:**

```bash
call-home
```

**Example:**

```bash
hostname(config) # call-home
```

Enters call-home configuration mode.

**Step 2**

**Command:**

```bash
alert-group {all | configuration | diagnostic | environment | inventory | syslog}
```

**Example:**

```bash
ciscoasa(cfg-call-home)# alert-group all
```

Enables the specified Smart Call Home group. Use the keyword **all** to enable all alert groups. By default, all alert groups are enabled.

**Step 3**

**Command:**

```bash
profile profile-name
```

**Example:**

```bash
hostname(cfg-call-home)# profile profile1
```

Enters the profile configuration submode for the specified destination profile.

**Step 4**

**Command:**

```bash
subscribe-to-alert-group configuration {periodic (daily hh:mm | monthly date hh:mm | weekly day hh:mm)}
```

**Example:**

```bash
hostname(cfg-call-home-profile)#
subsctibe-to-alert-group configuration periodic weekly Wednesday 23:30
```

Subscribes this destination profile to the configuration alert group. The configuration alert group can be configured for periodic notification, as described in the “Subscribing to Alert Groups” section on page 55-9.

To subscribe to all available alert groups, use the **subscribe-to-alert-group all** command.
Testing Call Home Communications

You can test Smart Call Home communications by sending messages manually using two command types. To send a user-defined Smart Call Home test message, use the `call-home test` command. To send a specific alert group message, use the `call-home send` command.

These sections describe Smart Call Home communication:

- Sending a Smart Call Home Test Message Manually, page 55-12
- Sending a Smart Call Home Alert Group Message Manually, page 55-12
- Sending the Output of a Command, page 55-12
Sending a Smart Call Home Test Message Manually

To manually send a Smart Call Home test message, perform this task:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>call-home test [test-message] profile profile-name</code></td>
<td>Sends a test message using a profile configuration.</td>
</tr>
</tbody>
</table>

Example:
```
hostname# call-home test [testing123] profile profile1
```

Sending a Smart Call Home Alert Group Message Manually

To manually trigger a Call Home alert group message, perform this task:

<table>
<thead>
<tr>
<th>Step 1</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`call-home send alert-group {inventory</td>
<td>configuration</td>
</tr>
</tbody>
</table>

Example:
```
hostname# call-home send alert-group inventory
```

Sending the Output of a Command

You can use the `call-home send` command to execute a CLI command and e-mail the command output to Cisco or to an e-mail address that you specify.

When sending the output of a command, the following guidelines apply:

- The specified CLI command can be any run command, including commands for all modules.
- If you specify an e-mail address, the command output is sent to that address. If no e-mail address is specified, the output is sent to Cisco TAC. The e-mail is sent in log text format with the service number, if specified, in the subject line.
- The service number is required only if no e-mail address is specified or if a Cisco TAC e-mail address is specified.

To execute a CLI command and e-mail the command output, perform this task:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>call-home send cli command [email email]</code></td>
<td>Sends command output to an e-mail address.</td>
</tr>
</tbody>
</table>

Example:
```
hostname# call-home send cli command email username@example.com
```
Optional Configuration Procedures

This section includes the following topics:

- Configuring Smart Call Home Customer Contact Information, page 55-13
- Configuring the Mail Server, page 55-15
- Configuring Call Home Traffic Rate Limiting, page 55-15
- Destination Profile Management, page 55-16

Configuring Smart Call Home Customer Contact Information

Obtain the following customer contact information to configure this task:

- E-mail address (required)
- Phone number (optional)
- Street address (optional)
- Contract ID (optional)
- Customer name (optional)
- Customer ID (optional)
- Site ID (optional)

To configure customer contact information, perform this task:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 call-home</td>
<td>Enters call home configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# call-home</td>
<td></td>
</tr>
<tr>
<td>Step 2 contact-email-addr</td>
<td>Configures the mandatory customer contact e-mail address (if you have</td>
</tr>
<tr>
<td>email-address</td>
<td>not already done so). The email-address should be the Cisco.com ID</td>
</tr>
<tr>
<td>Example:</td>
<td>account that is associated with the device.</td>
</tr>
<tr>
<td>ciscoasa(cfg-call-home)#</td>
<td></td>
</tr>
<tr>
<td>contact-email-addr</td>
<td></td>
</tr>
<tr>
<td><a href="mailto:username@example.com">username@example.com</a></td>
<td></td>
</tr>
<tr>
<td>Step 3 phone-number</td>
<td>Specifies a customer phone number.</td>
</tr>
<tr>
<td>phone-number-string</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>ciscoasa(cfg-call-home)#</td>
<td></td>
</tr>
<tr>
<td>phone-number 800555122</td>
<td></td>
</tr>
<tr>
<td>Step 4 street-address</td>
<td>Specifies the customer address, which is a free-format string that can</td>
</tr>
<tr>
<td>street-address</td>
<td>be up to 255 characters long.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>ciscoasa(cfg-call-home)#</td>
<td></td>
</tr>
<tr>
<td>street-address &quot;1234 Any</td>
<td></td>
</tr>
<tr>
<td>Street, Any city, Any state,</td>
<td></td>
</tr>
<tr>
<td>12345&quot;</td>
<td></td>
</tr>
</tbody>
</table>
This example shows the configuration of contact information:

```
hostname# configure terminal
hostname(config)# call-home
ciscoasa(config-call-home)# contact-email-addr username@example.com
    phone-number 8005551122
    street-address "1234 Any Street, Any city, Any state, 12345"
ciscoasa(config-call-home)# contact-name contactname1234
ciscoasa(config-call-home)# customer-id customer1234
ciscoasa(config-call-home)# site-id site1234
ciscoasa(config-call-home)# contract-id contract1234
```
Configuring Anonymous Reporting and Smart Call Home

Configuring the Mail Server

We recommend that you use HTTPS for message transport, as it is the most secure. However, you can configure an e-mail destination for Smart Call Home and then configure the mail server to use the e-mail message transport.

To configure the mail server, perform this task:

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>call-home</td>
<td>Enters call home configuration mode.</td>
</tr>
</tbody>
</table>

Example:

```
hostname(config)# call-home
```

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>mail-server ip-address</td>
<td>Specifies the SMTP mail server. Customers can specify up to five mail servers. At least one mail server is required for using e-mail transport for Smart Call Home messages. The lower the number, the higher the priority of the mail server. The ip-address option can be an IPv4 or IPv6 mail server address.</td>
</tr>
</tbody>
</table>

Example:

```
ciscoasa(cfg-call-home)# mail-server 10.10.1.1 smtp.example.com priority 1
```

Configuring Call Home Traffic Rate Limiting

You can configure this optional setting to specify the number of messages that Smart Call Home sends per minute.

To configure Smart Call Home traffic rate limiting, perform this task:

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>call-home</td>
<td>Enters call home configuration mode.</td>
</tr>
</tbody>
</table>

Example:

```
hostname(config)# call-home
```

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>rate-limit msg-count</td>
<td>Specifies the number of messages that Smart Call Home can send per minute. The default value is 10 messages per minute.</td>
</tr>
</tbody>
</table>

Example:

```
ciscoasa(cfg-call-home)# rate-limit 5
```
This example shows how to configure Smart Call Home traffic rate limiting:

```
hostname# configure terminal
hostname(config)# call-home
ciscoasa(cfg-call-home)# rate-limit 5
```

## Destination Profile Management

These sections describe destination profile management:
- Configuring a Destination Profile, page 55-16
- Activating and Deactivating a Destination Profile, page 55-17
- Copying a Destination Profile, page 55-18
- Renaming a Destination Profile, page 55-18

### Configuring a Destination Profile

To configure a destination profile for e-mail or for HTTP, perform this task:

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td><code>call-home</code></td>
<td>Enters call home configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
<td>hostname(config)# call-home</td>
</tr>
<tr>
<td>Step 2</td>
<td><code>profile profile-name</code></td>
<td>Enters the profile configuration mode for the specified destination profile. If the specified destination profile does not exist, it is created.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
<td>hostname(cfg-call-home)# profile newprofile</td>
</tr>
<tr>
<td>Step 3</td>
<td>`destination {email address</td>
<td>http url}</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
<td>hostname(cfg-call-home-profile)# destination address email <a href="mailto:username@example.com">username@example.com</a></td>
</tr>
<tr>
<td></td>
<td></td>
<td>hostname(cfg-call-home-profile)# destination preferred-msg-format long-text</td>
</tr>
</tbody>
</table>
Activating and Deactivating a Destination Profile

Smart Call Home destination profiles are automatically activated when you create them. If you do not want to use a profile right away, you can deactivate the profile.

To activate or deactivate a destination profile, perform this task:

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>call-home</td>
<td>Enters call home configuration mode.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>hostname(config)# call-home</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>profile profile-name</td>
<td>Enters the profile configuration mode. Creates, edits, or deletes a profile, which can be up to 20 characters long.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>hostname(cfg-call-home)# profile newprofile</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>active</td>
<td>Enables or disables a profile. By default, a new profile is enabled when it is created.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ciscoasa(cfg-call-home-profile)# active</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>no active</td>
<td>Disables the destination profile.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ciscoasa(cfg-call-home-profile)# no active</td>
<td></td>
</tr>
</tbody>
</table>

This example shows how to activate a destination profile:

```
hostname# configure terminal
ciscoasa(config)# call-home
ciscoasa(cfg-call-home)# profile newprofile
ciscoasa(cfg-call-home-profile)# active
ciscoasa(cfg-call-home)# end
```

This example shows how to deactivate a destination profile:

```
hostname# configure terminal
ciscoasa(config)# call-home
ciscoasa(cfg-call-home)# profile newprofile
ciscoasa(cfg-call-home-profile)# no active
ciscoasa(cfg-call-home)# end
```
Copying a Destination Profile
To create a new destination profile by copying an existing profile, perform this task:

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>call-home</td>
<td>Enters call home configuration mode.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>hostname(config)# call-home</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>profile profilename</td>
<td>Specifies the profile to copy.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ciscoasa(cfg-call-home)# profile newprofile</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>copy profile src-profile-name dest-profile-name</td>
<td>Copies the content of an existing profile (src-profile-name, which can be up to 23 characters long) to a new profile (dest-profile-name, which can be up to 23 characters long).</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ciscoasa(cfg-call-home)# copy profile profile1 profile2</td>
<td></td>
</tr>
</tbody>
</table>

This example shows how to copy an existing profile:

hostname# configure terminal
hostname(config)# call-home
ciscoasa(cfg-call-home)# profile newprofile
ciscoasa(cfg-call-home-profile)# copy profile profile1 profile2

Renaming a Destination Profile
To change the name of an existing profile, perform this task:

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>call-home</td>
<td>Enters call home configuration mode.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>hostname(config)# call-home</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>profile profilename</td>
<td>Specifies the profile to rename.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ciscoasa(cfg-call-home)# profile newprofile</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>rename profile src-profile-name dest-profile-name</td>
<td>Changes the name of an existing profile, the src-profile-name (an existing profile name can be up to 23 characters long), and the dest-profile-name (a new profile name can be up to 23 characters long).</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ciscoasa(cfg-call-home)# rename profile profile1 profile2</td>
<td></td>
</tr>
</tbody>
</table>

This example shows how to rename an existing profile:

hostname# configure terminal
hostname(config)# call-home
ciscoasa(cfg-call-home)# profile newprofile
Monitoring Smart Call Home

To monitor the Smart Call Home feature, enter one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show call-home detail</td>
<td>Shows the current Smart Call Home detail configuration.</td>
</tr>
<tr>
<td>show call-home mail-server status</td>
<td>Shows the current mail server status.</td>
</tr>
<tr>
<td>show call-home profile (profile name</td>
<td>all)</td>
</tr>
<tr>
<td>show call-home registered-module status [all]</td>
<td>Shows the registered module status.</td>
</tr>
<tr>
<td>show call-home statistics</td>
<td>Shows call-home detail status.</td>
</tr>
<tr>
<td>show call-home</td>
<td>Shows the current Smart Call Home configuration.</td>
</tr>
<tr>
<td>show running-config call-home</td>
<td>Shows the current Smart Call Home running configuration.</td>
</tr>
<tr>
<td>show smart-call-home alert-group</td>
<td>Shows the current status of Smart Call Home alert groups.</td>
</tr>
</tbody>
</table>

Configuration Example for Smart Call Home

The following example shows how to configure the Smart Call Home feature:

```plaintext
hostname (config)# service call-home
hostname (config)# call-home
hostname (cfg-call-home)# contact-email-addr customer@mail.server
hostname (cfg-call-home)# profile CiscoTAC-1
hostname (cfg-call-home-profile)# destination address http https://example.cisco.com/its/service/example/services/ExampleService
hostname (cfg-call-home-profile)# destination address email callhome@example.com
hostname (cfg-call-home-profile)# destination transport-method http
hostname (cfg-call-home-profile)# subscribe-to-alert-group inventory periodic monthly
hostname (cfg-call-home-profile)# subscribe-to-alert-group configuration periodic monthly
hostname (cfg-call-home-profile)# subscribe-to-alert-group environment
hostname (cfg-call-home-profile)# subscribe-to-alert-group diagnostic
hostname (cfg-call-home-profile)# subscribe-to-alert-group telemetry periodic daily
```
Feature History for Anonymous Reporting and Smart Call Home

Table 55-2 lists each feature change and the platform release in which it was implemented. ASDM is backwards-compatible with multiple platform releases, so the specific ASDM release in which support was added is not listed.

Table 55-2  Feature History for Anonymous Reporting and Smart Call Home

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Smart Call Home</td>
<td>8.2(2)</td>
<td>The Smart Call Home feature offers proactive diagnostics and real-time alerts on the ASASM, and provides higher network availability and increased operational efficiency. We introduced or modified the following commands: active (call home), call-home, call-home send alert-group, call-home test, contact-email-addr, customer-id (call home), destination (call home), profile, rename profile, service call-home, show call-home, show call-home detail, show smart-call-home alert-group, show call-home profile, show call-home statistics, show call-home mail-server status, show running-config call-home, show call-home registered-module status all, site-id, street-address, subscribe-to-alert-group all, subscribe-to-alert-group configuration, subscribe-to-alert-group diagnostic, subscribe-to-alert-group environment, subscribe-to-alert-group inventory, subscribe-to-alert-group syslog.</td>
</tr>
<tr>
<td>Anonymous Reporting</td>
<td>8.2(5)/8.4(2)</td>
<td>Customers can help to improve the ASA platform by enabling Anonymous Reporting, which allows Cisco to securely receive minimal error and health information from a device. We introduced the following commands: call-home reporting anonymous, call-home test reporting anonymous.</td>
</tr>
</tbody>
</table>
CHAPTER 56

Managing Software and Configurations

This chapter describes how to manage the ASASM software and configurations and includes the following sections:

- Managing the Flash File System, page 56-1
- Downloading Software or Configuration Files to Flash Memory, page 56-2
- Configuring the Application Image and ASDM Image to Boot, page 56-4
- Configuring the File to Boot as the Startup Configuration, page 56-5
- Deleting Files from a USB Drive on the ASA 5500-X Series, page 56-5
- Performing Zero Downtime Upgrades for Failover Pairs, page 56-6
- Backing Up Configuration Files or Other Files, page 56-8
- Configuring Auto Update Support, page 56-16
- Downgrading Your Software, page 56-19

Managing the Flash File System

This section includes the following topics:

- Viewing Files in Flash Memory, page 56-1
- Deleting Files from Flash Memory, page 56-2

Viewing Files in Flash Memory

You can view files in flash memory and see information about files as follows:

- To view files in flash memory, enter the following command:

  
  hostname# dir [disk0: | disk1:]

  Enter disk0: for the internal flash memory. The disk1: keyword represents the external flash memory. The internal flash memory is the default.

  For example:

  hostname# dir

  Directory of disk0:/
  500 -rw- 4958208 22:56:20 Nov 29 2004 cdisk.bin
To view extended information about a specific file, enter the following command:

```plaintext
hostname# show file information [path://]filename
```

The default path is the root directory of the internal flash memory (disk0://).

For example:

```plaintext
hostname# show file information cdisk.bin
```

```
disk0:/cdisk.bin:
  type is image (XXX) {}
  file size is 4976640 bytes version 7.0(1)
```

The file size listed is for example only.

### Deleting Files from Flash Memory

You can remove files from flash memory that you no longer need. To delete a file from flash memory, enter the following command:

```plaintext
hostname# delete disk0: filename
```

By default, the file is deleted from the current working directory if you do not specify a path. You may use wildcards when deleting files. You are prompted with the filename to delete, and then you must confirm the deletion.

### Downloading Software or Configuration Files to Flash Memory

You can download application images, ASDM images, configuration files, and other files to the internal flash memory or, for the ASASM, to the external flash memory from a TFTP, FTP, SMB, HTTP, or HTTPS server.

**Note**

You cannot have two files with the same name but with different letter case in the same directory in flash memory. For example, if you attempt to download the file, Config.cfg, to a location that contains the file, config.cfg, you receive the following error message:

```plaintext
%Error opening disk0:/Config.cfg (File exists).
```

This section includes the following topics:

- Downloading a File to a Specific Location, page 56-3
- Downloading a File to the Startup or Running Configuration, page 56-3
Chapter 56      Managing Software and Configurations

Chapter 56      Managing Software and Configurations

Downloading Software or Configuration Files to Flash Memory

This section describes how to download the application image, ASDM software, a configuration file, or any other file that needs to be downloaded to flash memory. To download a file to the running or startup configuration, see the “Downloading a File to the Startup or Running Configuration” section on page 56-3.

For information about installing the Cisco SSL VPN client, see the Cisco AnyConnect VPN Client Administrator Guide. For information about installing Cisco Secure Desktop on the ASASM, see the Cisco Secure Desktop Configuration Guide for Cisco ASA 5500 Series Administrators.

To configure the ASASM to use a specific application image or ASDM image if you have more than one installed, or have installed them in external flash memory, see the “Configuring the Application Image and ASDM Image to Boot” section on page 56-4.

To configure the ASASM to use a specific configuration as the startup configuration, see the “Configuring the File to Boot as the Startup Configuration” section on page 56-5.

For multiple context mode, you must be in the system execution space.

To download a file to flash memory, see the following commands for each download server type:

- To copy from a TFTP server, enter the following command:
  ```
  hostname# copy tftp://server[/path]/filename {disk0:/ | disk1:/}[path/]filename
  ```

- To copy from an FTP server, enter the following command:
  ```
  hostname# copy ftp://[user[:password]@]server[/path]/filename {disk0:/ | disk1:/}[path/]filename
  ```

- To copy from an HTTP or HTTPS server, enter the following command:
  ```
  hostname# copy http[s]://[user[:password]@]server[:port][/path]/filename {disk0:/ | disk1:/}[path/]filename
  ```

- To copy from an SMB server, enter the following command:
  ```
  hostname# copy smb://[user[:password]@]server[/path]/filename {disk0:/ | disk1:/}[path/]filename
  ```

- To use secure copy, first enable secure shell (SSH), and then enter the following command:
  ```
  hostname# ssh scopy enable
  ```
  From a Linux client, enter the following command:
  ```
  scp -v -pw password filename username@asa_address
  ```
  The -v is for verbose, and if -pw is not specified, you will be prompted for a password.

Downloads a File to a Specific Location

You can download a text file to the running or startup configuration from a TFTP, FTP, SMB, or HTTP(S) server, or from the flash memory.

To copy a file to the startup configuration or running configuration, enter one of the following commands for the appropriate download server:

```
hostname# copy tftp://server[/path]/filename {disk0:/ | disk1:/}[path/]filename
```

```
hostname# copy ftp://[user[:password]@]server[/path]/filename {disk0:/ | disk1:/}[path/]filename
```

```
hostname# copy http[s]://[user[:password]@]server[:port][/path]/filename {disk0:/ | disk1:/}[path/]filename
```

```
hostname# copy smb://[user[:password]@]server[/path]/filename {disk0:/ | disk1:/}[path/]filename
```

```
hostname# ssh scopy enable
```
When you copy a configuration to the running configuration, you merge the two configurations. A merge adds any new commands from the new configuration to the running configuration. If the configurations are the same, no changes occur. If commands conflict or if commands affect the running of the context, then the effect of the merge depends on the command. You might get errors, or you might have unexpected results.

- To copy from a TFTP server, enter the following command:

  `hostname# copy tftp://server[/path]/filename {startup-config | running-config}`

- To copy from an FTP server, enter the following command:

  `hostname# copy ftp://[user[:password@]server[/path]/filename {startup-config | running-config}`

- To copy from an HTTP or HTTPS server, enter the following command:

  `hostname# copy http[s]://[user[:password@]server[:port][/path]/filename {startup-config | running-config}`

- To copy from an SMB server, enter the following command:

  `hostname# copy smb://[user[:password@]server[/path]/filename {startup-config | running-config}`

- To copy from flash memory, enter the following command:

  `hostname# copy {disk0: | disk1:}[path/]filename {startup-config | running-config}`

For example, to copy the configuration from a TFTP server, enter the following command:

`hostname# copy tftp://209.165.200.226/configs/startup.cfg startup-config`

To copy the configuration from an FTP server, enter the following command:

`hostname# copy ftp://admin:letmein@209.165.200.227/configs/startup.cfg startup-config`

To copy the configuration from an HTTP server, enter the following command:

`hostname# copy http://209.165.200.228/configs/startup.cfg startup-config`

### Configuring the Application Image and ASDM Image to Boot

By default, the ASASM boots the first application image that it finds in internal flash memory. It also boots the first ASDM image it finds in internal flash memory, or if one does not exist in this location, then in external flash memory. If you have more than one image, you should specify the image that you want to boot. For the ASDM image, if you do not specify the image to boot, even if you have only one image installed, then the ASASM inserts the `asdm image` command into the running configuration. To avoid problems with Auto Update (if configured), and to avoid the image search at each startup, you should specify the ASDM image that you want to boot in the startup configuration.

To configure the application image to boot, enter the following command:

`hostname(config)# boot system url`

where `url` can be one of the following:

- `{disk0: | disk1:}[path/]filename`
• `tftp://[user[:password]@]server[:port]/[path/]filename`

**Note** The TFTP option is only supported for the ASASM.

You can enter up to four `boot system` command entries to specify different images to boot from in order; the ASASM boots the first image it finds. Only one `boot system tftp` command can be configured, and it must be the first one configured.

**Note** If the ASASM is stuck in a cycle of constant booting, you can reboot the ASASM into ROMMON mode. For more information about the ROMMON mode, see the “Using the ROM Monitor to Load a Software Image” section on page 57-11.

To configure the ASDM image to boot, enter the following command:

```
hostname(config)# asdm image {disk0:/ | disk1:}[path/]filename
```

### Configuring the File to Boot as the Startup Configuration

By default, the ASASM boots from a startup configuration that is a hidden file. You can alternatively set any configuration to be the startup configuration by entering the following command:

```
hostname(config)# boot config {disk0:/ | disk1:}[path/]filename
```

### Deleting Files from a USB Drive on the ASA 5500-X Series

When you delete a file from a USB drive (accessed as disk1:, for example), then the USB is moved to the other slot (from bottom to top, or top to bottom), and the file reappears. With this type of online insertion removal, to make sure that the file is actually deleted and no longer appears when you enter the `show disk1:` command, enter the following command:

```
hostname# eject disk1:
```
Performing Zero Downtime Upgrades for Failover Pairs

The two units in a failover configuration should have the same major (first number) and minor (second number) software version. However, you do not need to maintain version parity on the units during the upgrade process; you can have different versions on the software running on each unit and still maintain failover support. To ensure long-term compatibility and stability, we recommend upgrading both units to the same version as soon as possible.

Table 56-1 shows the supported scenarios for performing zero-downtime upgrades on a failover pair.

Table 56-1 Zero-Downtime Upgrade Support

<table>
<thead>
<tr>
<th>Type of Upgrade</th>
<th>Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maintenance Release</td>
<td>You can upgrade from any maintenance release to any other maintenance release within a minor release. For example, you can upgrade from 7.0(1) to 7.0(4) without first installing the maintenance releases in between.</td>
</tr>
<tr>
<td>Minor Release</td>
<td>You can upgrade from a minor release to the next minor release. You cannot skip a minor release. For example, you can upgrade from 7.0(1) to 7.1(1). Upgrading from 7.0(1) directly to 7.2(1) is not supported for zero-downtime upgrades; you must first upgrade to 7.1(1).</td>
</tr>
<tr>
<td>Major Release</td>
<td>You can upgrade from the last minor release of the previous version to the next major release. For example, you can upgrade from 7.2(1) to 8.0(1), assuming that 7.2(1) is the last minor version in the 7.x release series.</td>
</tr>
</tbody>
</table>

Note: Zero downtime upgrades are possible, even when feature configuration is migrated, for example, from 8.2.x to 8.3.x.

For more details about upgrading the software on a failover pair, see the following topics:

- Upgrading an Active/Standby Failover Configuration, page 56-6
- Upgrading an Active/Active Failover Configuration, page 56-7

Upgrading an Active/Standby Failover Configuration

To upgrade two units in an Active/Standby failover configuration, perform the following steps:

Step 1 Download the new software to both units, and specify the new image to load with the `boot system` command (see the “Configuring the Application Image and ASDM Image to Boot” section on page 56-4).

Step 2 Reload the standby unit to boot the new image by entering the following command on the active unit:

```
active# failover reload-standby
```

Step 3 When the standby unit has finished reloading, and is in the Standby Ready state, force the active unit to fail over to the standby unit by entering the following command on the active unit.
Performing Zero Downtime Upgrades for Failover Pairs

Note

Use the `show failover` command to verify that the standby unit is in the Standby Ready state.

```plaintext
active# no failover active
```

**Step 4**
Reload the former active unit (now the new standby unit) by entering the following command:

```plaintext
newstandby# reload
```

**Step 5**
When the new standby unit has finished reloading and is in the Standby Ready state, return the original active unit to active status by entering the following command:

```plaintext
newstandby# failover active
```

Upgrading an Active/Active Failover Configuration

To upgrade two units in an Active/Active failover configuration, perform the following steps:

**Step 1**
Download the new software to both units, and specify the new image to load with the `boot system` command (see the “Configuring the Application Image and ASDM Image to Boot” section on page 56-4).

**Step 2**
Make both failover groups active on the primary unit by entering the following command in the system execution space of the primary unit:

```plaintext
primary# failover active
```

**Step 3**
Reload the secondary unit to boot the new image by entering the following command in the system execution space of the primary unit:

```plaintext
primary# failover reload-standby
```

**Step 4**
When the secondary unit has finished reloading, and both failover groups are in the Standby Ready state on that unit, make both failover groups active on the secondary unit by using the following command in the system execution space of the primary unit:

```plaintext
primary# no failover active
```

Note

Use the `show failover` command to verify that both failover groups are in the Standby Ready state on the secondary unit.

```plaintext
primary# no failover active
```

**Step 5**
Make sure that both failover groups are in the Standby Ready state on the primary unit, and then reload the primary unit using the following command:

```plaintext
primary# reload
```

**Step 6**
If the failover groups are configured with the `preempt` command, they automatically become active on their designated unit after the preempt delay has passed. If the failover groups are not configured with the `preempt` command, you can return them to active status on their designated units using the `failover active group` command.
Backing Up Configuration Files or Other Files

This section includes the following topics:

- Backing up the Single Mode Configuration or Multiple Mode System Configuration, page 56-8
- Backing Up a Context Configuration or Other File in Flash Memory, page 56-8
- Backing Up a Context Configuration within a Context, page 56-9
- Copying the Configuration from the Terminal Display, page 56-9
- Backing Up Additional Files Using the Export and Import Commands, page 56-9
- Using a Script to Back Up and Restore Files, page 56-10

Backing up the Single Mode Configuration or Multiple Mode System Configuration

In single context mode or from the system configuration in multiple mode, you can copy the startup configuration or running configuration to an external server or to the local flash memory as follows:

- To copy to a TFTP server, enter the following command:
  ```
  hostname# copy (startup-config | running-config) tftp://server[/path]/filename
  ```
- To copy to a FTP server, enter the following command:
  ```
  hostname# copy (startup-config | running-config) ftp://[user[:password]@]server[/path]/filename
  ```
- To copy to local flash memory, enter the following command:
  ```
  hostname# copy (startup-config | running-config) {flash:// | disk0:// | disk1://}[path/]filename
  ```

  **Note** Be sure that the destination directory exists. If it does not exist, first create the directory using the `mkdir` command.

Backing Up a Context Configuration or Other File in Flash Memory

Copy context configurations or other files that are on the local flash memory by entering one of the following commands in the system execution space:

- To copy to a TFTP server, enter the following command:
  ```
  hostname# copy disk[0 | 1]:[/path/]filename tftp://server[/path]/filename
  ```
- To copy to a FTP server, enter the following command:
  ```
  hostname# copy disk[0 | 1]:[/path/]filename ftp://[user[:password]@]server[/path]/filename
  ```
- To copy to an SMB file-system, enter the following command:
  ```
  hostname# copy disk[0 | 1]:[/path/]filename smb://[user[:password]@]server[/path]/filename
  ```
To copy from the ASASM using HTTPS, enter the following URL in your browser:

https://ASA_IP/disk{0 | 1}/filename

To copy to local flash memory, enter the following command:

hostname# copy disk{0 | 1}:[/path/]filename disk{0 | 1}:[/path/]newfilename

Note: Be sure that the destination directory exists. If it does not exist, first create the directory using the mkdir command.

### Backing Up a Context Configuration within a Context

In multiple context mode, from within a context, you can perform the following backups:

- To copy the running configuration to the startup configuration server (connected to the admin context), enter the following command:

  hostname/contexta# copy running-config startup-config

- To copy the running configuration to a TFTP server connected to the context network, enter the following command:

  hostname/contexta# copy running-config tftp:/server[/path/]/filename

### Copying the Configuration from the Terminal Display

To print the configuration to the terminal, enter the following command:

hostname# show running-config

Copy the output from this command, and then paste the configuration into a text file.

### Backing Up Additional Files Using the Export and Import Commands

Additional files essential to your configuration might include the following:

- Files that you import using the `import webvpn` command. Currently, these files include customizations, URL lists, web content, plug-ins, and language translations.
- DAP policies (dap.xml).
- CSD configurations (data.xml).
- Digital keys and certificates.
- Local CA user database and certificate status files.

The CLI lets you back up and restore individual elements of your configuration using the `export` and `import` commands.

To back up these files, for example, those files that you imported with the `import webvpn` command or certificates, perform the following steps:

1. Run the applicable `show` command(s) as follows:
hostname # show import webvpn plug-in
ica
rdp
ssh, telnet
vnc

**Step 2** Run the `export` command for the file that you want to back up (in this example, the rdp file):
hostname # export webvpn plug-in protocol rdp tftp://tftpserver/backupfilename

---

**Using a Script to Back Up and Restore Files**

You can use a script to back up and restore the configuration files on your ASASM, including all extensions that you import via the `import webvpn` CLI, the CSD configuration XML files, and the DAP configuration XML file. For security reasons, we do not recommend that you perform automated backups of digital keys and certificates or the local CA key.

This section provides instructions for doing so and includes a sample script that you can use as is or modify as your environment requires. The sample script is specific to a Linux system. To use it for a Microsoft Windows system, you need to modify it using the logic of the sample.

---

**Note** The existing CLI lets you back up and restore individual files using the `copy`, `export`, and `import` commands. It does not, however, have a facility that lets you back up all ASASM configuration files in one operation. Running the script facilitates the use of multiple CLIs.

This section includes the following topics:

- Prerequisites, page 56-10
- Running the Script, page 56-10
- Sample Script, page 56-11

**Prerequisites**

To use a script to back up and restore an ASASM configuration, first perform the following tasks:

- Install Perl with an Expect module.
- Install an SSH client that can reach the ASASM.
- Install a TFTP server to send files from the ASASM to the backup site.

Another option is to use a commercially available tool. You can put the logic of this script into such a tool.

**Running the Script**

To run a backup-and-restore script, perform the following steps:

**Step 1** Download or cut-and-paste the script file to any location on your system.

**Step 2** At the command line, enter `Perl scriptname`, where `scriptname` is the name of the script file.

**Step 3** Press Enter.
Step 4  The system prompts you for values for each option. Alternatively, you can enter values for the options when you enter the Perl scriptname command before you press Enter. Either way, the script requires that you enter a value for each option.

Step 5  The script starts running, printing out the commands that it issues, which provides you with a record of the CLIs. You can use these CLIs for a later restore, which is particularly useful if you want to restore only one or two files.

Sample Script

#!/usr/bin/perl
#Function: Backup/restore configuration/extensions to/from a TFTP server.
#Description: The objective of this script is to show how to back up configurations/extensions before the backup/restore command is developed.
# It currently backs up the running configuration, all extensions imported via "import webvpn" command, the CSD configuration XML file, and the DAP configuration XML file.
#Requirements: Perl with Expect, SSH to the ASA, and a TFTP server.
#Usage: backupasa -option option_value
#   -h: ASA hostname or IP address
#   -u: User name to log in via SSH
#   -w: Password to log in via SSH
#   -e: The Enable password on the security appliance
#   -p: Global configuration mode prompt
#   -s: Host name or IP address of the TFTP server to store the configurations
#   -r: Restore with an argument that specifies the file name. This file is produced during backup.
#If you don't enter an option, the script will prompt for it prior to backup.
#Make sure that you can SSH to the ASA.

use Expect;
use Getopt::Std;

#global variables
$options=();
$restore = 0; #does backup by default
$restore_file = ''; 
$asa = '';
$storage = '';
$user = '';
$password = '';
$enable = '';
$prompt = '';
$date = `date +%F';
chop($date);
my $exp = new Expect();
getopts("h:u:p:w:e:s:r:",\%options);
do process_options();
do login($exp);
do enable($exp);
if ($restore) {
do restore($exp,$restore_file);
}
else {
$restore_file = "$prompt-restore-$date.cli";
open(OUT,">$restore_file") or die "Can't open $restore_file\n";
do running_config($exp);
do lang_trans($exp);
do customization($exp);
do plugin($exp);
do url_list($exp);
do webcontent($exp);
do dap($exp);
do cnsd($exp);
close(OUT);
}
do finish($exp);

sub enable {
$obj = shift;
$obj->send("enable\n");
unless ($obj->expect(15, 'Password:')) {
  print "timed out waiting for Password:\n";
}
$obj->send("$enable\n");
unless ($obj->expect(15, "$prompt#")) {
  print "timed out waiting for $prompt#\n";
}
}

sub lang_trans {
$obj = shift;
$obj->clear_accum();
$obj->send("show import webvpn translation-table\n");
$obj->expect(15, "$prompt#" );
$output = $obj->before();
@items = split(/\n+/,$output);
for (@items) {
  s/\s+/; s/\s+$//;
  next if /show import/ or /Translation Tables/;
  next unless (\^\s+.+$/);
  ($lang, $transtable) = split(\s+,$_);
  $cli = "export webvpn translation-table $transtable language $lang
$storage/$prompt-$date-$transtable-$lang.po";
  $ocli = $cli;
  $ocli =~ s/^export/import/;
  print "$ocli\n";
  print OUT "$ocli\n";
  $obj->send("$cli\n");
  $obj->expect(15, "$prompt#" );
}
}

sub running_config {
$obj = shift;
$obj->clear_accum();
$cli = "copy /noconfirm running-config $storage/$prompt-$date.cfg";
print "$cli\n";
$obj->send("$cli\n");
$obj->expect(15, "$prompt#" );
}

sub customization {
$obj = shift;
$obj->clear_accum();
$obj->send("show import webvpn customization\n");
$obj->expect(15, "$prompt#" );
$output = $obj->before();
@items = split(/\n+/,$output);
for (@items) {
    chop;
    next if /"Template/ or /show import/ or /\s*$/;
    $cli = "export webvpn customization $_ $storage/$prompt-$date-cust-$_.xml";
    $ocli = $cli;
    $ocli = s/"export/import/;
    print "$cli\n";
    print OUT "$cli\n";
    $obj->send("$cli\n");
    $obj->expect(15, "$prompt#" );
}

sub plugin {
    $obj = shift;
    $obj->clear_accum();
    $obj->send("show import webvpn plug-in\n");
    $obj->expect(15, "$prompt#" );
    $output = $obj->before();
    @items = split(/\n+, $output);
    for (@items) {
        chop;
        next if /"Template/ or /show import/ or /\s*$/ or /No bookmarks/;
        $cli = "export webvpn plug-in protocol $_ $storage/$prompt-$date-plugin-$_.jar";
        $ocli = $cli;
        $ocli = s/"export/import/;
        print "$cli\n";
        print OUT "$cli\n";
        $obj->send("$cli\n");
        $obj->expect(15, "$prompt#" );
    }
}

sub url_list {
    $obj = shift;
    $obj->clear_accum();
    $obj->send("show import webvpn url-list\n");
    $obj->expect(15, "$prompt#" );
    $output = $obj->before();
    @items = split(/\n+, $output);
    for (@items) {
        chop;
        next if /"Template/ or /show import/ or /\s*$/ or /No bookmarks/;
        $cli="export webvpn url-list $_ $storage/$prompt-$date-urllist-$_.xml";
        $ocli = $cli;
        $ocli = s/"export/import/;
        print "$cli\n";
        print OUT "$cli\n";
        $obj->send("$cli\n");
        $obj->expect(15, "$prompt#" );
    }
}

sub dap {
    $obj = shift;
    $obj->clear_accum();
    $obj->send("dir dap.xml\n");
    $obj->expect(15, "$prompt#" );
    $output = $obj->before();
    return 0 if($output =~ /Error/);
$cli="copy /noconfirm dap.xml $storage/$prompt-$date-dap.xml";
$scli="copy /noconfirm $storage/$prompt-$date-dap.xml disk0:/dap.xml";
print "$cli\n";
print OUT "$scli\n";
$obj->send("$cli\n");
$obj->expect(15, "$prompt#" );
}

sub csd {
    $obj = shift;
    $obj->clear_accum();
    $obj->send("dir sdesktop\n");
    $obj->expect(15, "$prompt#" );
    
    $output = $obj->before();
    return 0 if($output =~ /Error/);
    $cli="copy /noconfirm sdesktop/data.xml $storage/$prompt-$date-data.xml";
    $scli="copy /noconfirm $storage/$prompt-$date-data.xml disk0:/sdesktop/data.xml";
    print "$cli\n";
    print OUT "$scli\n";
    $obj->send("$cli\n");
    $obj->expect(15, "$prompt#" );
}

sub webcontent {
    $obj = shift;
    $obj->clear_accum();
    $obj->send("show import webvpn webcontent\n");
    $obj->expect(15, "$prompt#" );
    $output = $obj->before();
    @items = split(/\n+/, $output);
    for (@items) {
        s/\s+//;
        s/\s+$//;
        next if /show import/ or /No custom/;
        next unless (/^\+.+\+$/);
        ($url, $type) = split(/\s+/, $_);
        $turl = $url;
        $turl =~ s/\+/\-/;
        $cli = "export webvpn webcontent $url $storage/$prompt-$date-$turl";
        $scli = $cli;
        $scli = "$cli/"; export/import/; print "$scli\n";
        print OUT "$scli\n";
        $obj->send("$scli\n");
        $obj->expect(15, "$prompt#" );
    }
}

sub login {
    $obj = shift;
    $obj->raw_pty(1);
    $obj->log_stdout(0); #turn off console logging.
    $obj->spawn("/usr/bin/ssh $user@$asa") or die "can't spawn ssh\n";
    unless ($obj->expect(15, "password:" )) {
        die "timeout waiting for password:\n";
    }
    
    $obj->send("$password\n");
unless ($obj->expect(15, "$prompt>") ) {
  die "timeout waiting for $prompt>\n";
}

sub finish {
  $obj = shift;
  $obj->hard_close();
  print "\n\n";
}

sub restore {
  $obj = shift;
  my $file = shift;
  my $output;
  open(IN, "$file") or die "can't open $file\n";
  while (<IN>) {
    $obj->send("$_");
    $obj->expect(15, "$prompt# ");
    $output = $obj->before();
    print "$output\n";
  }
  close(IN);
}

sub process_options {
  if (defined($options{s})) {
    $tstr= $options{s};
    $storage = "tftp://$tstr";
  }
  else {
    print "Enter TFTP host name or IP address:";
    chop($tstr=<>);
    $storage = "tftp://$tstr";
  }
  if (defined($options{h})) {
    $asa = $options{h};
  }
  else {
    print "Enter ASA host name or IP address:";
    chop($asa=<>);
  }
  if (defined($options{u})) {
    $user= $options{u};
  }
  else {
    print "Enter user name:";
    chop($user=<>);
  }
  if (defined($options{w})) {
    $password= $options{w};
  }
  else {
    print "Enter password:";
    chop($password=<>);
  }
  if (defined($options{p})) {
    $prompt= $options{p};
  }
  else {
    print "Enter ASA prompt:";
  }
}
Configuring Auto Update Support

Auto Update is a protocol specification that allows an Auto Update Server to download configurations and software images to many ASASMs and can provide basic monitoring of the ASASMs from a central location.

The ASASM can be configured as either a client or a server. As an Auto Update client, it periodically polls the Auto Update Server for updates to software images and configuration files. As an Auto Update Server, it issues updates for ASASMs configured as Auto Update clients.

Note

Auto Update is supported in single context mode only.

This section includes the following topics:

- Configuring Communication with an Auto Update Server, page 56-16
- Configuring Client Updates as an Auto Update Server, page 56-18
- Viewing Auto Update Status, page 56-19

Configuring Communication with an Auto Update Server

To configure the ASASM as an Auto Update client, perform the following steps:

Step 1

To specify the URL of the Auto Update Server, enter the following command:

```
hostname(config)# auto-update server url [source interface] [verify-certificate]
```

where `url` has the following syntax:

http[s]://[user:password@]server_ip[:port]/pathname

SSL is used when https is specified. The `user` and `password` arguments of the URL are used for basic authentication when logging in to the server. If you use the `write terminal`, `show configuration` or `show tech-support` commands to view the configuration, the user and password are replaced with ‘********’.

The default port is 80 for HTTP and 443 for HTTPS.
The source interface keyword and argument specify which interface to use when sending requests to the Auto Update Server. If you specify the same interface specified by the management-access command, the Auto Update requests travel over the same IPsec VPN tunnel used for management access.

The verify-certificate keyword verifies the certificate returned by the Auto Update Server.

Step 2  (Optional) To identify the device ID to send when communicating with the Auto Update Server, enter the following command:

```
hostname(config)# auto-update device-id {hardware-serial | hostname | ipaddress [if-name] | mac-address [if-name] | string text}
```

The identifier used is determined by specifying one of the following parameters:

- The hardware-serial argument specifies the ASASM serial number.
- The hostname argument specifies the ASASM hostname.
- The ipaddress keyword specifies the IP address of the specified interface. If the interface name is not specified, it uses the IP address of the interface used to communicate with the Auto Update Server.
- The mac-address keyword specifies the MAC address of the specified interface. If the interface name is not specified, it uses the MAC address of the interface used to communicate with the Auto Update Server.
- The string keyword specifies the specified text identifier, which cannot include white space or the characters ",", ",", >, & and ?.

Step 3  (Optional) To specify how often to poll the Auto Update Server for configuration or image updates, enter the following command:

```
hostname(config)# auto-update poll-period poll-period [retry-count [retry-period]]
```

The poll-period argument specifies how often (in minutes) to check for an update. The default is 720 minutes (12 hours).

The retry-count argument specifies how many times to try reconnecting to the server if the first attempt fails. The default is zero.

The retry-period argument specifies how long to wait (in minutes) between retries. The default is five minutes.

Step 4  (Optional) To schedule a specific time for the ASASM to poll the Auto Update Server, enter the following command:

```
hostname(config)# auto-update poll-at days-of-the-week time [randomize minutes] [retry_count [retry_period]]
```

The days-of-the-week argument is any single day or combination of days: Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, and Sunday. Other possible values are daily (Monday through Sunday), weekdays (Monday through Friday), and weekends (Saturday and Sunday).

The time argument specifies the time in the format HH:MM at which to start the poll. For example, 8:00 is 8:00 a.m. and 20:00 is 8:00 p.m.

The randomize minutes keyword and argument specify the period to randomize the poll time following the specified start time. The range is from 1 to 1439 minutes.

The retry_count argument specifies how many times to try reconnecting to the Auto Update Server if the first attempt fails. The default is zero.

The retry_period argument specifies how long to wait between connection attempts. The default is five minutes. The range is from 1 to 35791 minutes.
Step 5  (Optional) If the Auto Update Server has not been contacted for a certain period of time, entering the following command causes it to stop passing traffic:

```
hostname(config)# auto-update timeout period
```

The `period` argument specifies the timeout period in minutes between 1 and 35791. The default is to never time out (zero minutes). To restore the default, enter the `no` form of this command.

Use the `auto-update timeout` command to be sure that the ASASM has the most recent image and configuration. This condition is reported with system log message 201008.

In the following example, an ASASM is configured to poll an Auto Update Server with the IP address 209.165.200.224, at port number 1742, from the outside interface, with certificate verification.

The ASASM is also configured to use the hostname as the device ID and to poll an Auto Update Server every Friday and Saturday night at a random time between 10:00 p.m. and 11:00 p.m. On a failed polling attempt, the ASASM will try to reconnect to the Auto Update Server ten times, and will wait three minutes between attempts at reconnecting, as shown in the following example:

```
hostname(config)# auto-update server
https://jcrichton:farscape@209.165.200.224:1742/management source outside verify-certificate
hostname(config)# auto-update device-id hostname
hostname(config)# auto-update poll-at Friday Saturday 22:00 randomize 60 2 10
```

### Configuring Client Updates as an Auto Update Server

Entering the `client-update` command enables updates for ASASMs configured as Auto Update clients and lets you specify the type of software component (ASDM or boot image), the type or family of ASASM, revision numbers to which the update applies, and a URL or IP address from which to obtain the update.

To configure the ASASM as an Auto Update Server, perform the following steps:

**Step 1**  To enable client update, enter the following command:

```
hostname(config)# client-update enable
```

**Step 2**  Configure the following parameters for the `client-update` command that you want to apply to the ASASMs:

```
client-update { component { asdm | image } | device-id dev_string | family family_name | type type } url url-string rev-nums rev-nums
```

The `component { asdm | image }` parameter specifies the software component, either ASDM or the boot image of the ASASM.

The `device-id dev_string` parameter specifies a unique string that the Auto Update client uses to identify itself. The maximum length is 63 characters.

The `family family_name` parameter specifies the family name that the Auto Update client uses to identify itself. It can be asa, pix, or a text string with a maximum length of seven characters.

The `rev-nums rev-nums` parameter specifies the software or firmware images for this client. Enter up to four, in any order, separated by commas.
The `type` parameter specifies the type of clients to notify of a client update. Because this command is also used to update Windows clients, the list of clients includes several Windows operating systems. The ASASMs in the list may include the following:

- asa5505: Cisco 5505 ASASM
- asa5510: Cisco 5510 ASASM
- asa5520: Cisco 5520 ASASM
- asa5540: Cisco 5540 ASASM

The `url url-string` parameter specifies the URL for the software/firmware image. This URL must point to a file appropriate for this client. For all Auto Update clients, you must use the protocol "http://" or "https://" as the prefix for the URL.

Configure the parameters for the client update that you want to apply to all ASASMs of a particular type. That is, specify the type of ASASM and the URL or IP address from which to get the updated image. In addition, you must specify a revision number. If the revision number of the remote ASASM matches one of the specified revision numbers, there is no need to update the client, and the update is ignored.

To configure a client update for Cisco 5520 ASASMs, enter the following command:

```
hostname(config)# client-update type asa5520 component asdm url http://192.168.1.114/aus/asdm601.bin rev-nums 8.0(1)
```

**Viewing Auto Update Status**

To view the Auto Update status, enter the following command:

```
hostname(config)# show auto-update
```

The following is sample output from the `show auto-update` command:

```
hostname(config)# show auto-update
Server: https://*******@209.165.200.224:1742/management.cgi?1276
Certificate will be verified
Poll period: 720 minutes, retry count: 2, retry period: 5 minutes
Timeout: none
Device ID: host name [corporate]
Next poll in 4.93 minutes
Last poll: 11:36:46 PST Tue Nov 13 2004
Last PDM update: 23:36:46 PST Tue Nov 12 2004
```

**Downgrading Your Software**

When you upgrade to Version 8.3, your configuration is migrated. The old configuration is automatically stored in flash memory. For example, when you upgrade from Version 8.2(1) to 8.3(1), the old 8.2(1) configuration is stored in flash memory in a file called `8_2_1_0_startup_cfg.sav`.

**Note**

You must manually restore the old configuration before downgrading.
This section describes how to downgrade and includes the following topics:

- Information About Activation Key Compatibility, page 56-20
- Performing the Downgrade, page 56-20

**Information About Activation Key Compatibility**

Your activation key remains compatible if you upgrade to the latest version from any previous version. However, you might have issues if you want to maintain downgrade capability:

- **Downgrading to Version 8.1 or earlier versions**—After you upgrade, if you activate additional feature licenses that were introduced before Version 8.2, the activation key continues to be compatible with earlier versions if you downgrade. However, if you activate feature licenses that were introduced in Version 8.2 or later versions, the activation key is not backward compatible. If you have an incompatible license key, see the following guidelines:
  - If you previously entered an activation key in an earlier version, the ASASM uses that key (without any of the new licenses you activated in Version 8.2 or later versions).
  - If you have a new system and do not have an earlier activation key, you need to request a new activation key compatible with the earlier version.

- **Downgrading to Version 8.2 or earlier versions**—Version 8.3 introduced more robust time-based key usage as well as failover license changes:
  - If you have more than one time-based activation key active, when you downgrade, only the most recently activated time-based key can be active. Any other keys are made inactive.
  - If you have mismatched licenses on a failover pair, downgrading will disable failover. Even if the keys are matching, the license used will no longer be a combined license.

**Performing the Downgrade**

To downgrade from Version 8.3, perform the following steps:

**Detailed Steps**

**Step 1**

Enter the following command:

```
hostname(config)# downgrade [/noconfirm] old_image_url old_config_url [activation-key old_key]
```

Where the `/noconfirm` option downgrades without prompting. The `image_url` is the path to the old image on disk0, disk1, tftp, ftp, or smb. The `old_config_url` is the path to the saved, premigration configuration (by default, this configuration was saved on disk0). If you need to revert to a pre-8.3 activation key, you can enter the old activation key.

This command is a shortcut for completing the following functions:

1. Clearing the boot image configuration (`clear configure boot`).
2. Setting the boot image to be the old image (`boot system`).
3. (Optional) Entering a new activation key (`activation-key`).
4. Saving the running configuration to startup (`write memory`). This action sets the BOOT environment variable to the old image, so when you reload, the old image is loaded.
5. Copying the old configuration to the startup configuration (**copy old_config_url startup-config**).
6. Reloading (**reload**).

For example:

```
hostname(config)# downgrade /noconfirm disk0:/asa821-k8.bin disk0:/8_2_1_0_startupCfg.sav
```
Troubleshooting

This chapter describes how to troubleshoot the ASASM and includes the following sections:

- Testing Your Configuration, page 57-1
- Reloading the ASASM, page 57-8
- Performing Password Recovery, page 57-8
- Using the ROM Monitor to Load a Software Image, page 57-11
- Erasing the Flash File System, page 57-12
- Other Troubleshooting Tools, page 57-13
- Common Problems, page 57-14

Testing Your Configuration

This section describes how to test connectivity for the single mode ASASM or for each security context, how to ping the ASASM interfaces, and how to allow hosts on one interface to ping through to hosts on another interface.

We recommend that you only enable pinging and debugging messages during troubleshooting. When you are done testing the ASASM, follow the steps in the “Disabling the Test Configuration” section on page 57-7.

This section includes the following topics:

- Enabling ICMP Debugging Messages and Syslog Messages, page 57-2
- Pinging ASASM Interfaces, page 57-3
- Passing Traffic Through the ASASM, page 57-5
- Disabling the Test Configuration, page 57-7
- Determining Packet Routing with Traceroute, page 57-7
- Tracing Packets with Packet Tracer, page 57-7
- Handling TCP Packet Loss, page 57-8
Enabling ICMP Debugging Messages and Syslog Messages

Debugging messages and syslog messages can help you troubleshoot why your pings are not successful. The ASASM only shows ICMP debugging messages for pings to the ASASM interfaces, and not for pings through the ASASM to other hosts. To enable debugging and syslog messages, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td>debug icmp trace</td>
<td>Shows ICMP packet information for pings to the ASASM interfaces.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# debug icmp trace</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td>logging monitor debug</td>
<td>Sets syslog messages to be sent to Telnet or SSH sessions.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# logging monitor debug</td>
<td></td>
</tr>
<tr>
<td><strong>Note</strong></td>
<td>You can alternately use the logging buffer debug command to send log messages to a buffer, and then view them later using the show logging command.</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td></td>
</tr>
<tr>
<td>terminal monitor</td>
<td>Sends the syslog messages to a Telnet or SSH session.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# terminal monitor</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td></td>
</tr>
<tr>
<td>logging on</td>
<td>Enables syslog message generation.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# logging on</td>
<td></td>
</tr>
</tbody>
</table>

To enable ICMP inspection to the default global policy, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td>policy-map name</td>
<td>Configures the policy map and attach the action to the class of traffic.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# policy-map global_policy</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td>class classmap_name</td>
<td>Assigns a class map to the policy map so that you can assign actions to the class map traffic.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config-pmap)# class inspection_default</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td></td>
</tr>
<tr>
<td>inspect icmp</td>
<td>Enables ICMP inspection.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# inspect icmp</td>
<td></td>
</tr>
</tbody>
</table>
Examples

The following example shows a successful ping from an external host (209.165.201.2) to the ASASM outside interface (209.165.201.1):

```
hostname(config)# debug icmp trace
Inbound ICMP echo reply (len 32 id 1 seq 256) 209.165.201.1 > 209.165.201.2
Outbound ICMP echo request (len 32 id 1 seq 512) 209.165.201.2 > 209.165.201.1
Inbound ICMP echo reply (len 32 id 1 seq 512) 209.165.201.1 > 209.165.201.2
Outbound ICMP echo request (len 32 id 1 seq 768) 209.165.201.2 > 209.165.201.1
Inbound ICMP echo reply (len 32 id 1 seq 768) 209.165.201.1 > 209.165.201.2
Outbound ICMP echo request (len 32 id 1 seq 1024) 209.165.201.2 > 209.165.201.1
Inbound ICMP echo reply (len 32 id 1 seq 1024) 209.165.201.1 > 209.165.201.2
```

The output shows the ICMP packet length (32 bytes), the ICMP packet identifier (1), and the ICMP sequence number (the ICMP sequence number starts at 0, and is incremented each time that a request is sent).

Pinging ASASM Interfaces

To test whether the ASASM interfaces are up and running and that the ASASM and connected routers are operating correctly, you can ping the ASASM interfaces. To ping the ASASM interfaces, perform the following steps:

**Step 1**
Draw a diagram of your single-mode ASASM or security context that shows the interface names, security levels, and IP addresses.

*Note* Although this procedure uses IP addresses, the `ping` command also supports DNS names and names that are assigned to a local IP address with the `name` command.

The diagram should also include any directly connected routers and a host on the other side of the router from which you will ping the ASASM. You will use this information in this procedure and in the procedure in the “Passing Traffic Through the ASASM” section on page 57-5. (See Figure 57-1.)
Testing Your Configuration

Figure 57-1  Network Diagram with Interfaces, Routers, and Hosts

Step 2  Ping each ASASM interface from the directly connected routers. For transparent mode, ping the management IP address. This test ensures that the ASASM interfaces are active and that the interface configuration is correct.

A ping might fail if the ASASM interface is not active, the interface configuration is incorrect, or if a switch between the ASASM and a router is down (see Figure 57-2). In this case, no debug messages or syslog messages appear, because the packet never reaches the ASASM.

Figure 57-2  Ping Failure at the ASASM Interface

If the ping reaches the ASASM, and it responds, debugging messages similar to the following appear:

ICMP echo reply (len 32 id 1 seq 256) 209.165.201.1 > 209.165.201.2
ICMP echo request (len 32 id 1 seq 512) 209.165.201.2 > 209.165.201.1

If the ping reply does not return to the router, then a switch loop or redundant IP addresses may exist (see Figure 57-3).
Step 3

Ping each ASASM interface from a remote host. For transparent mode, ping the management IP address. This test checks whether the directly connected router can route the packet between the host and the ASASM, and whether the ASASM can correctly route the packet back to the host.

A ping might fail if the ASASM does not have a return route to the host through the intermediate router (see Figure 57-4). In this case, the debugging messages show that the ping was successful, but syslog message 110001 appears, indicating a routing failure.

Passing Traffic Through the ASASM

After you successfully ping the ASASM interfaces, make sure that traffic can pass successfully through the ASASM. For routed mode, this test shows that NAT is operating correctly, if configured. For transparent mode, which does not use NAT, this test confirms that the ASASM is operating correctly. If the ping fails in transparent mode, contact the Cisco TAC.

To ping between hosts on different interfaces, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><strong>access-list ICMPACL extended permit icmp any any</strong></td>
</tr>
<tr>
<td><strong>Example:</strong>:</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# access-list ICMPACL extended permit icmp any any</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><strong>access-group ICMPACL in interface interface_name</strong></td>
</tr>
<tr>
<td><strong>Example:</strong>:</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# access-group ICMPACL in interface inside</td>
<td></td>
</tr>
</tbody>
</table>

**Note** By default, when hosts access a lower security interface, all traffic is allowed through. However, to access a higher security interface, you need the preceding access list.
Step 3

```
class-map ICMP-CLASS
match access-list ICMPACL
policy-map ICMP-POLICY
class ICMP-CLASS
inspect icmp
service-policy ICMP-POLICY global
```

Example:
```
hostname(config)# class-map ICMP-CLASS
hostname(config-cmap)# match access-list ICMPACL
hostname(config)# policy-map ICMP-POLICY
hostname(config-pmap)# class ICMP-CLASS
hostname(config-pmap)# inspect icmp
hostname(config)# service-policy ICMP-POLICY global
```

Enables the ICMP inspection engine and ensures that ICMP responses may return to the source host.

For a host to access a lower security interface, you must enable ICMP inspection. However, to access a higher security interface, you must enable ICMP inspection and the preceding access list.

**Note**

Alternatively, you can also apply the ICMP access list to the destination interface to allow ICMP traffic back through the ASASM.

Step 4

```/logging on
```

Example:
```
hostname(config)# logging on
```

Enables syslog message generation.

If the ping succeeds, a syslog message appears to confirm the address translation for routed mode (305009 or 305011) and that an ICMP connection was established (302020). You can also enter either the `show xlate` or `show conns` command to view this information.

If the ping fails for transparent mode, contact Cisco TAC.

For routed mode, the ping might fail because NAT is not configured correctly (see Figure 57-5). In this case, a syslog message appears, showing that the NAT failed (305005 or 305006). If the ping is from an outside host to an inside host, and you do not have a static translation, the following syslog message appears:

```
%ASA-3-106010: deny inbound icmp.
```

**Note**
The ASASM only shows ICMP debugging messages for pings to the ASASM interfaces, and not for pings through the ASASM to other hosts.

![Figure 57-5 Ping Failure Because the ASASM is Not Translating Addresses](image-url)
Disabling the Test Configuration

After you complete your testing, disable the test configuration that allows ICMP to and through the ASASM and that prints debugging messages. If you leave this configuration in place, it can pose a serious security risk. Debugging messages also slow the ASASM performance.

To disable the test configuration, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>no debug icmp trace</td>
<td>Disables ICMP debugging messages.</td>
</tr>
<tr>
<td>no logging on</td>
<td>Disables logging.</td>
</tr>
<tr>
<td>no access-list ICMPACL</td>
<td>Removes the ICMPACL access list, and deletes the related access-group commands.</td>
</tr>
<tr>
<td>no service-policy ICMP-POLICY</td>
<td>(Optional) Disables the ICMP inspection engine.</td>
</tr>
</tbody>
</table>

Determining Packet Routing with Traceroute

You can trace the route of a packet using the traceroute feature, which is accessed with the traceroute command. A traceroute works by sending UDP packets to a destination on an invalid port. Because the port is not valid, the routers along the way to the destination respond with an ICMP Time Exceeded Message, and report that error to the ASASM.

Tracing Packets with Packet Tracer

The packet tracer tool provides packet tracing for packet sniffing and network fault isolation, as well as detailed information about the packets and how they are processed by the ASASM. If a configuration command did not cause the packet to drop, the packet tracer tool provides information about the cause in an easily readable manner.

In addition, you can trace the lifespan of a packet through the ASASM to see whether the packet is operating correctly with the packet tracer tool. This tool enables you to do the following:
- Debug all packet drops in a production network.
- Verify the configuration is working as intended.
- Show all rules applicable to a packet, along with the CLI commands that caused the rule addition.
- Show a time line of packet changes in a data path.
- Inject tracer packets into the data path.
- Search for an IPv4 or IPv6 address based on the user identity and the FQDN.

To trace packets, enter the following command:

```
hostname# packet-tracer input inside tcp 10.2.25.3 www 209.165.202.158 aol detailed
```

Provides detailed information about the packets and how they are processed by the ASASM. The example shows how to enable packet tracing from inside host 10.2.25.3 to external host 209.165.202.158, including detailed information.

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>reload</code></td>
<td>Restarts the ASASM.</td>
</tr>
</tbody>
</table>

In multiple mode, you can only reload from the system execution space.

---

### Handling TCP Packet Loss

To troubleshoot TCP packet loss, see the “Customizing the TCP Normalizer with a TCP Map” section on page 44-6 for more information.

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>reload</code></td>
<td>Restarts the ASASM.</td>
</tr>
</tbody>
</table>

---

### Reloading the ASASM

To reload the ASASM, enter the following command:

```
hostname (config)# reload
```

---

### Performing Password Recovery

This section describes how to recover passwords if you have forgotten them or you are locked out because of AAA settings, and how to disable password recovery for extra security. This section includes the following topics:

- Recovering Passwords for the ASASM, page 57-9
Performing Password Recovery

Disabling Password Recovery, page 57-10
Resetting the Password on the SSM Hardware Module, page 57-11

Recovering Passwords for the ASASM

To recover passwords for the ASASM, perform the following steps:

Step 1  Connect to the ASASM console port according to the instructions in “Accessing the ASA Services Module Command-Line Interface” section on page 3-1.

Step 2  Power off the ASASM, and then power it on.

Step 3  After startup, press the Escape key when you are prompted to enter ROMMON mode.

Step 4  To update the configuration register value, enter the following command:

    rommon #1> confreg 0x41
    Update Config Register (0x41) in NVRAM...

Step 5  To set the ASASM to ignore the startup configuration, enter the following command:

    rommon #1> confreg

The ASASM displays the current configuration register value, and asks whether you want to change it:

    Current Configuration Register: 0x00000041
    Configuration Summary:
        boot default image from Flash
        ignore system configuration

    Do you wish to change this configuration? y/n [n]: y

Step 6  Record the current configuration register value, so you can restore it later.

Step 7  At the prompt, enter Y to change the value.

The ASASM prompts you for new values.

Step 8  Accept the default values for all settings. At the prompt, enter Y.

Step 9  Reload the ASASM by entering the following command:

    rommon #2> boot
    Launching BootLoader...
    Boot configuration file contains 1 entry.
    Loading disk0:/asa800-226-k8.bin... Booting...Loading...

The ASASM loads the default configuration instead of the startup configuration.

Step 10 Access the privileged EXEC mode by entering the following command:

    hostname# enable

Step 11 When prompted for the password, press Enter.

The password is blank.

Step 12 Load the startup configuration by entering the following command:

    hostname# copy startup-config running-config

Step 13 Access the global configuration mode by entering the following command:

    hostname# configure terminal
Performing Password Recovery

Step 14 Change the passwords, as required, in the default configuration by entering the following commands:

```
hostname(config)# password password
hostname(config)# enable password password
hostname(config)# username name password password
```

Step 15 Load the default configuration by entering the following command:

```
hostname(config)# no config-register
```

The default configuration register value is 0x1. For more information about the configuration register, see the command reference.

Step 16 Save the new passwords to the startup configuration by entering the following command:

```
hostname(config)# copy running-config startup-config
```

Disabling Password Recovery

To disable password recovery to ensure that unauthorized users cannot use the password recovery mechanism to compromise the ASASM, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>no service password-recovery</code></td>
<td>Disables password recovery.</td>
</tr>
</tbody>
</table>

Example:

```
hostname (config)# no service password-recovery
```

On the ASASM, the `no service password-recovery` command prevents you from entering ROMMON mode with the configuration intact. When you enter ROMMON mode, the ASASM prompts you to erase all Flash file systems. You cannot enter ROMMON mode without first performing this erasure. If you choose not to erase the Flash file system, the ASASM reloads. Because password recovery depends on using ROMMON mode and maintaining the existing configuration, this erasure prevents you from recovering a password. However, disabling password recovery prevents unauthorized users from viewing the configuration or inserting different passwords. In this case, to restore the system to an operating state, load a new image and a backup configuration file, if available.

The `service password-recovery` command appears in the configuration file for information only. When you enter the command at the CLI prompt, the setting is saved in NVRAM. The only way to change the setting is to enter the command at the CLI prompt. Loading a new configuration with a different version of the command does not change the setting. If you disable password recovery when the ASASM is configured to ignore the startup configuration at startup (in preparation for password recovery), then the ASASM changes the setting to load the startup configuration as usual. If you use failover, and the standby unit is configured to ignore the startup configuration, then the same change is made to the configuration register when the `no service password recovery` command replicates to the standby unit.
Resetting the Password on the SSM Hardware Module

To reset the password to the default of “cisco” on the SSM hardware module, enter the following command:

```
Note
Make sure that the SSM hardware module is in the Up state and supports password reset.
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>hw-module module 1</td>
<td>Reset the password on module in slot 1? [confirm]</td>
</tr>
<tr>
<td>password-reset</td>
<td>Where ( l ) is the specified slot number on the SSM hardware module.</td>
</tr>
<tr>
<td>y</td>
<td>On the AIP SSM, entering this command reboots the hardware module.</td>
</tr>
<tr>
<td></td>
<td>The module is offline until the rebooting is finished. Enter the show</td>
</tr>
<tr>
<td></td>
<td>module command to monitor the module status. The AIP SSM supports this</td>
</tr>
<tr>
<td></td>
<td>command in version 6.0 and later.</td>
</tr>
<tr>
<td>hostname# y</td>
<td>On the CSC SSM, entering this command resets web services on the</td>
</tr>
<tr>
<td></td>
<td>hardware module after the password has been reset. You may lose</td>
</tr>
<tr>
<td></td>
<td>connection to ASDM or be logged out of the hardware module. The CSC</td>
</tr>
<tr>
<td></td>
<td>SSM supports this command in the most recent version of 6.3, dated</td>
</tr>
<tr>
<td></td>
<td>January 2010, and later releases.</td>
</tr>
</tbody>
</table>

Using the ROM Monitor to Load a Software Image

To load a software image to an ASASM from the ROM monitor mode using TFTP, perform the following steps:

**Step 1**
Connect to the ASASM console port according to the instructions in the “Accessing the ASA Services Module Command-Line Interface” section on page 3-1.

**Step 2**
Make sure that you reload the ASASM image.

**Step 3**
During startup, press the *Escape* key when you are prompted to enter ROMMON mode.

**Step 4**
In ROMMOM mode, define the interface settings to the ASASM, including the IP address, TFTP server address, gateway address, software image file, port, and VLAN, as follows:

```
rommon #1> ADDRESS=172.16.145.149
rommon #2> SERVER=172.16.171.125
rommon #3> GATEWAY=172.16.145.129
rommon #4> IMAGE=f1/asa851-smp-k8.bin
rommon #5> PORT=Data0
rommon #6> VLAN=1
Data0
```

```
Link is UP
MAC Address: 0012.d949.15b8
```

**Note**
Be sure that the connection to the network already exists.

**Step 5**
To validate your settings, enter the *set* command.

```
rommon #7> set
```
Erasing the Flash File System

To erase the flash file system, perform the following steps:

Step 1  Connect to the ASASM console port according to the instructions in “Accessing the ASA Services Module Command-Line Interface” section on page 3-1.
Step 2  Power off the ASASM, then power it on.
Step 3  During startup, press the **Escape** key when you are prompted to enter ROMMON mode.
Step 4  Enter the **erase** command, which overwrites all files and erases the file system, including hidden system files.

```
rommon #1> erase [disk0: | disk1: | flash:]
```

Other Troubleshooting Tools

The ASASM provides other troubleshooting tools that you can use. This section includes the following topics:

- Viewing Debugging Messages, page 57-13
- Capturing Packets, page 57-14
- Viewing the Crash Dump, page 57-14
- Coredump, page 57-14
- Monitoring Per-Process CPU Usage, page 57-14

Viewing Debugging Messages

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco TAC. Moreover, it is best to use **debug** commands during periods
Capturing Packets

Capturing packets is sometimes useful when troubleshooting connectivity problems or monitoring suspicious activity. We recommend that you contact the Cisco TAC if you want to use the packet capture feature. See the capture command in the command reference.

Viewing the Crash Dump

If the ASASM crashes, you can view the crash dump information. We recommend contacting Cisco TAC if you want to interpret the crash dump. See the show crashdump command in the command reference.

Coredump

A coredump is a snapshot of the running program when the program has terminated abnormally, or crashed. Coredumps are used to diagnose or debug errors and save a crash for future off-site analysis. Cisco TAC may request that users enable the coredump feature to troubleshoot application or system crashes on the ASASM. See the coredump command in the command reference.

Monitoring Per-Process CPU Usage

You can monitor the processes that run on the CPU. You can obtain information about the percentage of CPU that is used by a certain process. CPU usage statistics are sorted in descending order to display the highest consumer at the top. Also included is information about the load on the CPU per process, at 5 seconds, 1 minute, and 5 minutes before the log time. This information is updated automatically every 5 seconds to provide real-time statistics. You can use the show process cpu-usage sorted command to find a breakdown of the process-related load-to-CPU that is consumed by any configured contexts.

Common Problems

This section describes common problems with the ASASM, and how you might resolve them.

Symptom  The context configuration was not saved, and was lost when you reloaded.

Possible Cause  You did not save each context within the context execution space. If you are configuring contexts at the command line, you did not save the current context before you changed to the next context.

Recommended Action  Save each context within the context execution space using the copy start run command. Load the startup configuration as your active configuration. Then change the password and then enter the copy run start command. You cannot save contexts from the system execution space.
Symptom  You cannot make a Telnet or SSH connection to the ASASM interface.

Possible Cause  You did not enable Telnet or SSH to the ASASM.

Recommended Action  Enable Telnet or SSH to the ASASM according to the instructions in the “Configuring ASA Access for ASDM, Telnet, or SSH” section on page 34-1.

Symptom  You cannot ping the ASASM interface.

Possible Cause  You disabled ICMP to the ASASM.

Recommended Action  Enable ICMP to the ASASM for your IP address using the icmp command.

Symptom  You cannot ping through the ASASM, although the access list allows it.

Possible Cause  You did not enable the ICMP inspection engine or apply access lists on both the ingress and egress interfaces.

Recommended Action  Because ICMP is a connectionless protocol, the ASASM does not automatically allow returning traffic through. In addition to an access list on the ingress interface, you either need to apply an access list to the egress interface to allow replying traffic, or enable the ICMP inspection engine, which treats ICMP connections as stateful connections.

Symptom  Traffic does not pass between two interfaces on the same security level.

Possible Cause  You did not enable the feature that allows traffic to pass between interfaces at the same security level.

Recommended Action  Enable this feature according to the instructions in the “Allowing Same Security Level Communication” section on page 7-12.

Symptom  IPsec tunnels do not duplicate during a failover to the standby device.

Possible Cause  The switch port that the ASASM is plugged into is set to 10/100 instead of 1000.

Recommended Action  Set the switch port that the ASASM is plugged into to 1000.
P a r t  1 6

Reference
Using the Command-Line Interface

This appendix describes how to use the CLI on the ASASM and includes the following sections:

- Firewall Mode and Security Context Mode, page A-1
- Command Modes and Prompts, page A-2
- Syntax Formatting, page A-3
- Abbreviating Commands, page A-3
- Command-Line Editing, page A-3
- Command Completion, page A-4
- Command Help, page A-4
- Filtering show Command Output, page A-4
- Command Output Paging, page A-5
- Adding Comments, page A-5
- Text Configuration Files, page A-5
- Supported Character Sets, page A-8

Note

The CLI uses similar syntax and other conventions to the Cisco IOS CLI, but the ASASM operating system is not a version of Cisco IOS software. Do not assume that a Cisco IOS CLI command works with or has the same function on the ASASM.

Firewall Mode and Security Context Mode

The ASASM runs in a combination of the following modes:

- Transparent firewall or routed firewall mode
  The firewall mode determines if the ASASM runs as a Layer 2 or Layer 3 firewall.
- Multiple context or single context mode
  The security context mode determines if the ASASM runs as a single device or as multiple security contexts, which act like virtual devices.

Some commands are only available in certain modes.
Command Modes and Prompts

The ASASM CLI includes command modes. Some commands can only be entered in certain modes. For example, to enter commands that show sensitive information, you need to enter a password and enter a more privileged mode. Then, to ensure that configuration changes are not entered accidentally, you have to enter a configuration mode. All lower commands can be entered in higher modes, for example, you can enter a privileged EXEC command in global configuration mode.

Note

The various types of prompts are all default prompts and when configured, they can be different.

- When you are in the system configuration or in single context mode, the prompt begins with the hostname:
  hostname

- When printing the prompt string, the prompt configuration is parsed and the configured keyword values are printed in the order in which you have set the prompt command. The keyword arguments can be any of the following and in any order: hostname, domain, context, priority, state.
  asa(config)# prompt hostname context priority state

- When you are within a context, the prompt begins with the hostname followed by the context name:
  hostname/context

The prompt changes depending on the access mode:

- User EXEC mode
  User EXEC mode lets you see minimum ASASM settings. The user EXEC mode prompt appears as follows when you first access the ASASM:
    hostname>
    hostname/context>

- Privileged EXEC mode
  Privileged EXEC mode lets you see all current settings up to your privilege level. Any user EXEC mode command will work in privileged EXEC mode. Enter the enable command in user EXEC mode, which requires a password, to start privileged EXEC mode. The prompt includes the number sign (#):
    hostname#
    hostname/context#

- Global configuration mode
  Global configuration mode lets you change the ASASM configuration. All user EXEC, privileged EXEC, and global configuration commands are available in this mode. Enter the configure terminal command in privileged EXEC mode to start global configuration mode. The prompt changes to the following:
    hostname(config)#
    hostname/context(config)#

- Command-specific configuration modes
From global configuration mode, some commands enter a command-specific configuration mode. All user EXEC, privileged EXEC, global configuration, and command-specific configuration commands are available in this mode. For example, the `interface` command enters interface configuration mode. The prompt changes to the following:

```
hostname(config-if)#
hostname/context(config-if)#
```

### Syntax Formatting

Command syntax descriptions use the conventions listed in Table A-1.

<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>bold</strong></td>
<td>Bold text indicates commands and keywords that you enter literally as shown.</td>
</tr>
<tr>
<td><em>italics</em></td>
<td>Italic text indicates arguments for which you supply values.</td>
</tr>
<tr>
<td><code>[x]</code></td>
<td>Square brackets enclose an optional element (keyword or argument).</td>
</tr>
<tr>
<td>`{x</td>
<td>y}`</td>
</tr>
<tr>
<td><code>[x | y]</code></td>
<td>Square brackets enclosing keywords or arguments separated by a vertical bar indicate an optional choice.</td>
</tr>
<tr>
<td>`{x</td>
<td>y}`</td>
</tr>
<tr>
<td>`[x {y</td>
<td>z}]`</td>
</tr>
</tbody>
</table>

### Abbreviating Commands

You can abbreviate most commands down to the fewest unique characters for a command; for example, you can enter `wr` to view the configuration instead of entering the full command `write terminal`, or you can enter `en` to start privileged mode and `conf` to start configuration mode. In addition, you can enter `0` to represent `0.0.0.0`.

### Command-Line Editing

The ASASM uses the same command-line editing conventions as Cisco IOS software. You can view all previously entered commands with the `show history` command or individually with the up arrow or `^p` command. Once you have examined a previously entered command, you can move forward in the list with the down arrow or `^n` command. When you reach a command you wish to reuse, you can edit it or press the `Enter` key to start it. You can also delete the word to the left of the cursor with `^w`, or erase the line with `^u`.

The ASASM permits up to 512 characters in a command; additional characters are ignored.
Command Completion

To complete a command or keyword after entering a partial string, press the Tab key. The ASASM only completes the command or keyword if the partial string matches only one command or keyword. For example, if you enter s and press the Tab key, the ASASM does not complete the command because it matches more than one command. However, if you enter dis, the Tab key completes the disable command.

Command Help

Help information is available from the command line by entering the following commands:

- help command_name
  Shows help for the specific command.
- command_name ?
  Shows a list of arguments available.
- string? (no space)
  Lists the possible commands that start with the string.
- ? and +?
  Lists all commands available. If you enter ?, the ASASM shows only commands available for the current mode. To show all commands available, including those for lower modes, enter +?.

Note

If you want to include a question mark (?) in a command string, you must press Ctrl-V before typing the question mark so that you do not inadvertently invoke CLI help.

Filtering show Command Output

You can use the vertical bar (|) with any show command and include a filter option and filtering expression. The filtering is performed by matching each output line with a regular expression, similar to Cisco IOS software. By selecting different filter options you can include or exclude all output that matches the expression. You can also display all output beginning with the line that matches the expression.

The syntax for using filtering options with the show command is as follows:

hostname# show command | {include | exclude | begin | grep [-v]} regexp

In this command string, the first vertical bar (|) is the operator and must be included in the command. This operator directs the output of the show command to the filter. In the syntax diagram, the other vertical bars (|) indicate alternative options and are not part of the command.

The include option includes all output lines that match the regular expression. The grep option without -v has the same effect. The exclude option excludes all output lines that match the regular expression. The grep option with -v has the same effect. The begin option shows all the output lines starting with the line that matches the regular expression.
Replace \textit{regexp} with any Cisco IOS regular expression. The regular expression is not enclosed in quotes or double-quotes, so be careful with trailing white spaces, which will be taken as part of the regular expression.

When creating regular expressions, you can use any letter or number that you want to match. In addition, certain keyboard characters called \textit{metacharacters} have special meaning when used in regular expressions.

Use \texttt{Ctrl+V} to escape all of the special characters in the CLI, such as a question mark (\texttt{?}) or a tab. For example, type \texttt{d[Ctrl+V]?g} to enter \texttt{d?g} in the configuration.

For a list of metacharacters, see Table 12-2 on page 12-13.

### Command Output Paging

For commands such as \texttt{help} or \texttt{?}, \texttt{show}, \texttt{show xlate}, or other commands that provide long listings, you can determine if the information displays a screen and pauses, or lets the command run to completion. The \texttt{pager} command lets you choose the number of lines to display before the More prompt appears.

When paging is enabled, the following prompt appears:

\texttt{<--- More --->}

The More prompt uses syntax similar to the UNIX \texttt{more} command:

- To view another screen, press the \texttt{Space} bar.
- To view the next line, press the \texttt{Enter} key.
- To return to the command line, press the \texttt{q} key.

### Adding Comments

You can precede a line with a colon (\texttt{:}) to create a comment. However, the comment only appears in the command history buffer and not in the configuration. Therefore, you can view the comment with the \texttt{show history} command or by pressing an arrow key to retrieve a previous command, but because the comment is not in the configuration, the \texttt{write terminal} command does not display it.

### Text Configuration Files

This section describes how to format a text configuration file that you can download to the ASASM, and includes the following topics:

- \texttt{How Commands Correspond with Lines in the Text File}, page A-6
- \texttt{Command-Specific Configuration Mode Commands}, page A-6
- \texttt{Automatic Text Entries}, page A-7
- \texttt{Line Order}, page A-7
- \texttt{Commands Not Included in the Text Configuration}, page A-7
- \texttt{Passwords}, page A-7
- \texttt{Multiple Security Context Files}, page A-7
How Commands Correspond with Lines in the Text File

The text configuration file includes lines that correspond with the commands described in this guide. In examples, commands are preceded by a CLI prompt. The prompt in the following example is “hostname(config)#”:

```
hostname(config)# context a
```

In the text configuration file you are not prompted to enter commands, so the prompt is omitted:

```
context a
```

Command-Specific Configuration Mode Commands

Command-specific configuration mode commands appear indented under the main command when entered at the command line. Your text file lines do not need to be indented, as long as the commands appear directly following the main command. For example, the following unindented text is read the same as indented text:

```
interface gigabitethernet0/0
  nameif inside
interface gigabitethernet0/1
  nameif outside
```
Automatic Text Entries

When you download a configuration to the ASASM, it inserts some lines automatically. For example, the ASASM inserts lines for default settings or for the time the configuration was modified. You do not need to enter these automatic entries when you create your text file.

Line Order

For the most part, commands can be in any order in the file. However, some lines, such as ACEs, are processed in the order they appear, and the order can affect the function of the access list. Other commands might also have order requirements. For example, you must enter the nameif command for an interface first because many subsequent commands use the name of the interface. Also, commands in a command-specific configuration mode must directly follow the main command.

Commands Not Included in the Text Configuration

Some commands do not insert lines in the configuration. For example, a runtime command such as show running-config does not have a corresponding line in the text file.

Passwords

The login, enable, and user passwords are automatically encrypted before they are stored in the configuration. For example, the encrypted form of the password “cisco” might look like jMorNbK0514fadBh. You can copy the configuration passwords to another ASASM in its encrypted form, but you cannot unencrypt the passwords yourself.

If you enter an unencrypted password in a text file, the ASASM does not automatically encrypt it when you copy the configuration to the ASASM. The ASASM only encrypts it when you save the running configuration from the command line using the copy running-config startup-config or write memory command.

Multiple Security Context Files

For multiple security contexts, the entire configuration consists of the following multiple parts:

- The security context configurations
- The system configuration, which identifies basic settings for the ASASM, including a list of contexts
- The admin context, which provides network interfaces for the system configuration
  
  The system configuration does not include any interfaces or network settings for itself. Rather, when the system needs to access network resources (such as downloading the contexts from the server), it uses a context that is designated as the admin context.

Each context is similar to a single context mode configuration. The system configuration differs from a context configuration in that the system configuration includes system-only commands (such as a list of all contexts) while other typical commands are not present (such as many interface parameters).
**Supported Character Sets**

The ASASM CLI currently supports UTF-8 encoding only. UTF-8 is the particular encoding scheme for Unicode symbols, and has been designed to be compatible with an ASCII subset of symbols. ASCII characters are represented in UTF-8 as one-byte characters. All other characters are represented in UTF-8 as multibyte symbols.

The ASCII printable characters (0x20 to 0x7e) are fully supported. The printable ASCII characters are the same as ISO 8859-1. UTF-8 is a superset of ISO 8859-1, so the first 256 characters (0-255) are the same as ISO 8859-1. The ASASM CLI supports up to 255 characters (multibyte characters) of ISO 8859-1.
APPENDIX B

Addresses, Protocols, and Ports

This appendix provides a quick reference for IP addresses, protocols, and applications. This appendix includes the following sections:

- IPv4 Addresses and Subnet Masks, page B-1
- IPv6 Addresses, page B-5
- Protocols and Applications, page B-11
- TCP and UDP Ports, page B-11
- Local Ports and Protocols, page B-14
- ICMP Types, page B-15

IPv4 Addresses and Subnet Masks

This section describes how to use IPv4 addresses in the ASASM. An IPv4 address is a 32-bit number written in dotted-decimal notation: four 8-bit fields (octets) converted from binary to decimal numbers, separated by dots. The first part of an IP address identifies the network on which the host resides, while the second part identifies the particular host on the given network. The network number field is called the network prefix. All hosts on a given network share the same network prefix but must have a unique host number. In classful IP, the class of the address determines the boundary between the network prefix and the host number.

This section includes the following topics:

- Classes, page B-1
- Private Networks, page B-2
- Subnet Masks, page B-2

Classes

IP host addresses are divided into three different address classes: Class A, Class B, and Class C. Each class fixes the boundary between the network prefix and the host number at a different point within the 32-bit address. Class D addresses are reserved for multicast IP.

- Class A addresses (1.xxx.xxx.xxx through 126.xxx.xxx.xxx) use only the first octet as the network prefix.
IPv4 Addresses and Subnet Masks

• Class B addresses (128.0.xxx.xxx through 191.255.xxx.xxx) use the first two octets as the network prefix.
• Class C addresses (192.0.0.xxx through 223.255.255.xxx) use the first three octets as the network prefix.

Because Class A addresses have 16,777,214 host addresses, and Class B addresses 65,534 hosts, you can use subnet masking to break these huge networks into smaller subnets.

Private Networks

If you need large numbers of addresses on your network, and they do not need to be routed on the Internet, you can use private IP addresses that the Internet Assigned Numbers Authority (IANA) recommends (see RFC 1918). The following address ranges are designated as private networks that should not be advertised:

• 10.0.0.0 through 10.255.255.255
• 172.16.0.0 through 172.31.255.255
• 192.168.0.0 through 192.168.255.255

Subnet Masks

A subnet mask lets you convert a single Class A, B, or C network into multiple networks. With a subnet mask, you can create an extended network prefix that adds bits from the host number to the network prefix. For example, a Class C network prefix always consists of the first three octets of the IP address. But a Class C extended network prefix uses part of the fourth octet as well.

Subnet masking is easy to understand if you use binary notation instead of dotted decimal. The bits in the subnet mask have a one-to-one correspondence with the Internet address:

• The bits are set to 1 if the corresponding bit in the IP address is part of the extended network prefix.
• The bits are set to 0 if the bit is part of the host number.

Example 1: If you have the Class B address 129.10.0.0 and you want to use the entire third octet as part of the extended network prefix instead of the host number, then you must specify a subnet mask of 11111111.11111111.11111111.00000000. This subnet mask converts the Class B address into the equivalent of a Class C address, where the host number consists of the last octet only.

Example 2: If you want to use only part of the third octet for the extended network prefix, then you must specify a subnet mask like 11111111.11111111.11111000.00000000, which uses only 5 bits of the third octet for the extended network prefix.

You can write a subnet mask as a dotted-decimal mask or as a /bits (“slash bits”) mask. In Example 1, for a dotted-decimal mask, you convert each binary octet into a decimal number: 255.255.255.0. For a /bits mask, you add the number of 1s: /24. In Example 2, the decimal number is 255.255.248.0 and the /bits is /21.

You can also supernet multiple Class C networks into a larger network by using part of the third octet for the extended network prefix. For example, 192.168.0.0/20.

This section includes the following topics:

• Determining the Subnet Mask, page B-3
• Determining the Address to Use with the Subnet Mask, page B-3
Determining the Subnet Mask

To determine the subnet mask based on how many hosts you want, see Table B-1.

<table>
<thead>
<tr>
<th>Table B-1</th>
<th>Hosts, Bits, and Dotted-Decimal Masks</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Hosts 1</td>
</tr>
<tr>
<td>16,777,216</td>
<td>/8</td>
</tr>
<tr>
<td>65,536</td>
<td>/16</td>
</tr>
<tr>
<td>32,768</td>
<td>/17</td>
</tr>
<tr>
<td>16,384</td>
<td>/18</td>
</tr>
<tr>
<td>8192</td>
<td>/19</td>
</tr>
<tr>
<td>4096</td>
<td>/20</td>
</tr>
<tr>
<td>2048</td>
<td>/21</td>
</tr>
<tr>
<td>1024</td>
<td>/22</td>
</tr>
<tr>
<td>512</td>
<td>/23</td>
</tr>
<tr>
<td>256</td>
<td>/24</td>
</tr>
<tr>
<td>128</td>
<td>/25</td>
</tr>
<tr>
<td>64</td>
<td>/26</td>
</tr>
<tr>
<td>32</td>
<td>/27</td>
</tr>
<tr>
<td>16</td>
<td>/28</td>
</tr>
<tr>
<td>8</td>
<td>/29</td>
</tr>
<tr>
<td>4</td>
<td>/30</td>
</tr>
<tr>
<td>Do not use</td>
<td>/31</td>
</tr>
<tr>
<td>1</td>
<td>/32</td>
</tr>
</tbody>
</table>

1. The first and last number of a subnet are reserved, except for /32, which identifies a single host.

Determining the Address to Use with the Subnet Mask

The following sections describe how to determine the network address to use with a subnet mask for a Class C-size and a Class B-size network. This section includes the following topics:

- Class C-Size Network Address, page B-3
- Class B-Size Network Address, page B-4

Class C-Size Network Address

For a network between 2 and 254 hosts, the fourth octet falls on a multiple of the number of host addresses, starting with 0. For example, Table B-2 shows the 8-host subnets (/29) of 192.168.0.x.

<table>
<thead>
<tr>
<th>Table B-2</th>
<th>Class C-Size Network Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subnet with Mask /29 (255.255.255.248)</td>
<td>Address Range 1</td>
</tr>
<tr>
<td>192.168.0.0</td>
<td>192.168.0.0 to 192.168.0.7</td>
</tr>
<tr>
<td>192.168.0.8</td>
<td>192.168.0.8 to 192.168.0.15</td>
</tr>
</tbody>
</table>
Class B-Size Network Address

To determine the network address to use with the subnet mask for a network with between 254 and 65,534 hosts, you need to determine the value of the third octet for each possible extended network prefix. For example, you might want to subnet an address like 10.1.x.0, where the first two octets are fixed because they are used in the extended network prefix, and the fourth octet is 0 because all bits are used for the host number.

To determine the value of the third octet, follow these steps:

**Step 1**
Calculate how many subnets you can make from the network by dividing 65,536 (the total number of addresses using the third and fourth octet) by the number of host addresses you want.

For example, 65,536 divided by 4096 hosts equals 16.

Therefore, there are 16 subnets of 4096 addresses each in a Class B-size network.

**Step 2**
Determine the multiple of the third octet value by dividing 256 (the number of values for the third octet) by the number of subnets:

In this example, 256/16 = 16.

The third octet falls on a multiple of 16, starting with 0.

Therefore, Table B-3 shows the 16 subnets of the network 10.1.

<table>
<thead>
<tr>
<th>Subnet with Mask /20 (255.255.240.0)</th>
<th>Address Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.1.0.0</td>
<td>10.1.0.0 to 10.1.15.255</td>
</tr>
<tr>
<td>10.1.16.0</td>
<td>10.1.16.0 to 10.1.31.255</td>
</tr>
<tr>
<td>10.1.32.0</td>
<td>10.1.32.0 to 10.1.47.255</td>
</tr>
<tr>
<td>10.1.240.0</td>
<td>10.1.240.0 to 10.1.255.255</td>
</tr>
</tbody>
</table>

1. The first and last address of a subnet are reserved. In the first subnet example, you cannot use 10.1.0.0 or 10.1.15.255.
IPv6 Addresses

IPv6 is the next generation of the Internet Protocol after IPv4. It provides an expanded address space, a simplified header format, improved support for extensions and options, flow labeling capability, and authentication and privacy capabilities. IPv6 is described in RFC 2460. The IPv6 addressing architecture is described in RFC 3513.

This section describes the IPv6 address format and architecture and includes the following topics:

- IPv6 Address Format, page B-5
- IPv6 Address Types, page B-6
- IPv6 Address Prefixes, page B-10

Note

This section describes the IPv6 address format, the types, and prefixes. For information about configuring the ASASM to use IPv6, see the “Configuring IPv6 Addressing” section on page 7-8.

IPv6 Address Format

IPv6 addresses are represented as a series of eight 16-bit hexadecimal fields separated by colons (:) in the format: x:x:x:x:x:x:x:x. The following are two examples of IPv6 addresses:

- 2001:0DB8:0000:0000:0008:0800:200C:417A

Note

The hexadecimal letters in IPv6 addresses are not case-sensitive.

You do not need to include the leading zeros in an individual field of the address, but each field must contain at least one digit. So the example address 2001:0DB8:0000:0000:0008:0800:200C:417A can be shortened to 2001:0DB8:0:8:800:200C:417A by removing the leading zeros from the third through sixth fields from the left. The fields that contained all zeros (the third and fourth fields from the left) were shortened to a single zero. The fifth field from the left had the three leading zeros removed, leaving a single 8 in that field, and the sixth field from the left had the one leading zero removed, leaving 800 in that field.

It is common for IPv6 addresses to contain several consecutive hexadecimal fields of zeros. You can use two colons (::) to compress consecutive fields of zeros at the beginning, middle, or end of an IPv6 address (the colons represent the successive hexadecimal fields of zeros). Table B-4 shows several examples of address compression for different types of IPv6 address.

<table>
<thead>
<tr>
<th>Address Type</th>
<th>Standard Form</th>
<th>Compressed Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unicast</td>
<td>2001:0DB8:0:0:0:BA98:0:3210</td>
<td>2001:0DB8::BA98:0:3210</td>
</tr>
<tr>
<td>Multicast</td>
<td>FF01:0:0:0:0:0:101</td>
<td>FF01::101</td>
</tr>
<tr>
<td>Loopback</td>
<td>0:0:0:0:0:0:1</td>
<td>::1</td>
</tr>
<tr>
<td>Unspecified</td>
<td>0:0:0:0:0:0:0:0</td>
<td>::</td>
</tr>
</tbody>
</table>
IPv6 Addresses

Note

Two colons (::) can be used only once in an IPv6 address to represent successive fields of zeros.

An alternative form of the IPv6 format is often used when dealing with an environment that contains both IPv4 and IPv6 addresses. This alternative has the format x:x:x:x:x:y.y.y, where x represent the hexadecimal values for the six high-order parts of the IPv6 address and y represent decimal values for the 32-bit IPv4 part of the address (which takes the place of the remaining two 16-bit parts of the IPv6 address). For example, the IPv4 address 192.168.1.1 could be represented as the IPv6 address 0:0:0:0:0:FFFF:192.168.1.1 or ::FFFF:192.168.1.1.

IPv6 Address Types

The following are the three main types of IPv6 addresses:

- **Unicast**—A unicast address is an identifier for a single interface. A packet sent to a unicast address is delivered to the interface identified by that address. An interface may have more than one unicast address assigned to it.

- **Multicast**—A multicast address is an identifier for a set of interfaces. A packet sent to a multicast address is delivered to all addresses identified by that address.

- **Anycast**—An anycast address is an identifier for a set of interfaces. Unlike a multicast address, a packet sent to an anycast address is only delivered to the “nearest” interface, as determined by the measure of distances for the routing protocol.

Note

There are no broadcast addresses in IPv6. Multicast addresses provide the broadcast functionality.

This section includes the following topics:

- Unicast Addresses, page B-6
- Multicast Address, page B-8
- Anycast Address, page B-9
- Required Addresses, page B-10

Unicast Addresses

This section describes IPv6 unicast addresses. Unicast addresses identify an interface on a network node.

This section includes the following topics:

- Global Address, page B-7
- Site-Local Address, page B-7
- Link-Local Address, page B-7
- IPv4-Compatible IPv6 Addresses, page B-7
- Unspecified Address, page B-8
- Loopback Address, page B-8
- Interface Identifiers, page B-8
Global Address

The general format of an IPv6 global unicast address is a global routing prefix followed by a subnet ID followed by an interface ID. The global routing prefix can be any prefix not reserved by another IPv6 address type (see the “IPv6 Address Prefixes” section on page B-10, for information about the IPv6 address type prefixes).

All global unicast addresses, other than those that start with binary 000, have a 64-bit interface ID in the Modified EUI-64 format. See the “Interface Identifiers” section on page B-8, for more information about the Modified EUI-64 format for interface identifiers.

Global unicast address that start with the binary 000 do not have any constraints on the size or structure of the interface ID portion of the address. One example of this type of address is an IPv6 address with an embedded IPv4 address (see the “IPv4-Compatible IPv6 Addresses” section on page B-7).

Site-Local Address

Site-local addresses are used for addressing within a site. They can be used to address an entire site without using a globally unique prefix. Site-local addresses have the prefix FEC0::/10, followed by a 54-bit subnet ID, and end with a 64-bit interface ID in the modified EUI-64 format.

Site-local routers do not forward any packets that have a site-local address for a source or destination outside of the site. Therefore, site-local addresses can be considered private addresses.

Link-Local Address

All interfaces are required to have at least one link-local address. You can configure multiple IPv6 addresses per interfaces, but only one link-local address.

A link-local address is an IPv6 unicast address that can be automatically configured on any interface using the link-local prefix FE80::/10 and the interface identifier in modified EUI-64 format. Link-local addresses are used in the neighbor discovery protocol and the stateless autoconfiguration process. Nodes with a link-local address can communicate; they do not need a site-local or globally unique address to communicate.

Routers do not forward any packets that have a link-local address for a source or destination. Therefore, link-local addresses can be considered private addresses.

IPv4-Compatible IPv6 Addresses

There are two types of IPv6 addresses that can contain IPv4 addresses.

The first type is the IPv4-compatibly IPv6 address. The IPv6 transition mechanisms include a technique for hosts and routers to dynamically tunnel IPv6 packets over IPv4 routing infrastructure. IPv6 nodes that use this technique are assigned special IPv6 unicast addresses that carry a global IPv4 address in the low-order 32 bits. This type of address is termed an IPv4-compatible IPv6 address and has the format ::y.y.y.y, where y.y.y.y is an IPv4 unicast address.

Note

The IPv4 address used in the IPv4-compatible IPv6 address must be a globally unique IPv4 unicast address.

The second type of IPv6 address, which holds an embedded IPv4 address, is called the IPv4-mapped IPv6 address. This address type is used to represent the addresses of IPv4 nodes as IPv6 addresses. This type of address has the format ::FFFF:y.y.y.y, where y.y.y.y is an IPv4 unicast address.
IPv6 Addresses

Unspecified Address

The unspecified address, 0:0:0:0:0:0:0:0, indicates the absence of an IPv6 address. For example, a newly initialized node on an IPv6 network may use the unspecified address as the source address in its packets until it receives its IPv6 address.

Note

The IPv6 unspecified address cannot be assigned to an interface. The unspecified IPv6 addresses must not be used as destination addresses in IPv6 packets or the IPv6 routing header.

Loopback Address

The loopback address, 0:0:0:0:0:0:0:1, may be used by a node to send an IPv6 packet to itself. The loopback address in IPv6 functions the same as the loopback address in IPv4 (127.0.0.1).

Note

The IPv6 loopback address cannot be assigned to a physical interface. A packet that has the IPv6 loopback address as its source or destination address must remain within the node that created the packet. IPv6 routers do not forward packets that have the IPv6 loopback address as their source or destination address.

Interface Identifiers

Interface identifiers in IPv6 unicast addresses are used to identify the interfaces on a link. They need to be unique within a subnet prefix. In many cases, the interface identifier is derived from the interface link-layer address. The same interface identifier may be used on multiple interfaces of a single node, as long as those interfaces are attached to different subnets.

For all unicast addresses, except those that start with the binary 000, the interface identifier is required to be 64 bits long and to be constructed in the Modified EUI-64 format. The Modified EUI-64 format is created from the 48-bit MAC address by inverting the universal/local bit in the address and by inserting the hexadecimal number FFFE between the upper three bytes and lower three bytes of the of the MAC address.

For example, an interface with the MAC address of 00E0.b601.3B7A would have a 64-bit interface ID of 02E0:B6FF:FE01:3B7A.

Multicast Address

An IPv6 multicast address is an identifier for a group of interfaces, typically on different nodes. A packet sent to a multicast address is delivered to all interfaces identified by the multicast address. An interface may belong to any number of multicast groups.

An IPv6 multicast address has a prefix of FF00::/8 (1111 1111). The octet following the prefix defines the type and scope of the multicast address. A permanently assigned (well known) multicast address has a flag parameter equal to 0; a temporary (transient) multicast address has a flag parameter equal to 1. A multicast address that has the scope of a node, link, site, or organization, or a global scope has a scope parameter of 1, 2, 5, 8, or E, respectively. For example, a multicast address with the prefix FF02::/16 is a permanent multicast address with a link scope. Figure B-1 shows the format of the IPv6 multicast address.
IPv6 nodes (hosts and routers) are required to join the following multicast groups:

- The All Nodes multicast addresses:
  - FF01:: (interface-local)
  - FF02:: (link-local)
- The Solicited-Node Address for each IPv6 unicast and anycast address on the node:
  FF02:0:0:0:1:FFXX::XX/104, where XX:XXXX is the low-order 24-bits of the unicast or anycast address.

Note Solicited-Node addresses are used in Neighbor Solicitation messages.

IPv6 routers are required to join the following multicast groups:

- FF01::2 (interface-local)
- FF02::2 (link-local)
- FF05::2 (site-local)

Multicast address should not be used as source addresses in IPv6 packets.

Note There are no broadcast addresses in IPv6. IPv6 multicast addresses are used instead of broadcast addresses.

Anycast Address

The IPv6 anycast address is a unicast address that is assigned to more than one interface (typically belonging to different nodes). A packet that is routed to an anycast address is routed to the nearest interface having that address, the nearness being determined by the routing protocol in effect.

Anycast addresses are allocated from the unicast address space. An anycast address is simply a unicast address that has been assigned to more than one interface, and the interfaces must be configured to recognize the address as an anycast address.

The following restrictions apply to anycast addresses:

- An anycast address cannot be used as the source address for an IPv6 packet.
IPv6 Addresses

- An anycast address cannot be assigned to an IPv6 host; it can only be assigned to an IPv6 router.

Note

Anycast addresses are not supported on the ASASM.

Required Addresses

IPv6 hosts must, at a minimum, be configured with the following addresses (either automatically or manually):

- A link-local address for each interface
- The loopback address
- The All-Nodes multicast addresses
- A Solicited-Node multicast address for each unicast or anycast address

IPv6 routers must, at a minimum, be configured with the following addresses (either automatically or manually):

- The required host addresses
- The Subnet-Router anycast addresses for all interfaces for which it is configured to act as a router
- The All-Routers multicast addresses

IPv6 Address Prefixes

An IPv6 address prefix, in the format ipv6-prefix/prefix-length, can be used to represent bit-wise contiguous blocks of the entire address space. The IPv6-prefix must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons. The prefix length is a decimal value that indicates how many of the high-order contiguous bits of the address comprise the prefix (the network portion of the address). For example, 2001:0DB8:8086:6502::/32 is a valid IPv6 prefix.

The IPv6 prefix identifies the type of IPv6 address. Table B-5 shows the prefixes for each IPv6 address type.

Table B-5 IPv6 Address Type Prefixes

<table>
<thead>
<tr>
<th>Address Type</th>
<th>Binary Prefix</th>
<th>IPv6 Notation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unspecified</td>
<td>000...0 (128 bits)</td>
<td>::/128</td>
</tr>
<tr>
<td>Loopback</td>
<td>000...1 (128 bits)</td>
<td>::1/128</td>
</tr>
<tr>
<td>Multicast</td>
<td>11111111</td>
<td>FF00::/8</td>
</tr>
<tr>
<td>Link-Local (unicast)</td>
<td>1111110000000000</td>
<td>FE80::/10</td>
</tr>
<tr>
<td>Site-Local (unicast)</td>
<td>1111111111</td>
<td>FEC0::/10</td>
</tr>
<tr>
<td>Global (unicast)</td>
<td>All other addresses.</td>
<td></td>
</tr>
<tr>
<td>Anycast</td>
<td>Taken from the unicast address space.</td>
<td></td>
</tr>
</tbody>
</table>
Protocols and Applications

Table B-6 lists the protocol literal values and port numbers; either can be entered in ASASM commands.

<table>
<thead>
<tr>
<th>Literal</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ah</td>
<td>51</td>
<td>Authentication Header for IPv6, RFC 1826.</td>
</tr>
<tr>
<td>esp</td>
<td>50</td>
<td>Encapsulated Security Payload for IPv6, RFC 1827.</td>
</tr>
<tr>
<td>gre</td>
<td>47</td>
<td>Generic Routing Encapsulation.</td>
</tr>
<tr>
<td>icmp</td>
<td>1</td>
<td>Internet Control Message Protocol, RFC 792.</td>
</tr>
<tr>
<td>icmp6</td>
<td>58</td>
<td>Internet Control Message Protocol for IPv6, RFC 2463.</td>
</tr>
<tr>
<td>igmp</td>
<td>2</td>
<td>Internet Group Management Protocol, RFC 1112.</td>
</tr>
<tr>
<td>igrp</td>
<td>9</td>
<td>Interior Gateway Routing Protocol.</td>
</tr>
<tr>
<td>ip</td>
<td>0</td>
<td>Internet Protocol.</td>
</tr>
<tr>
<td>ipinip</td>
<td>4</td>
<td>IP-in-IP encapsulation.</td>
</tr>
<tr>
<td>ipsec</td>
<td>50</td>
<td>IP Security. Entering the ipsec protocol literal is equivalent to entering the esp protocol literal.</td>
</tr>
<tr>
<td>nos</td>
<td>94</td>
<td>Network Operating System (Novell’s NetWare).</td>
</tr>
<tr>
<td>ospf</td>
<td>89</td>
<td>Open Shortest Path First routing protocol, RFC 1247.</td>
</tr>
<tr>
<td>pcp</td>
<td>108</td>
<td>Payload Compression Protocol.</td>
</tr>
<tr>
<td>pim</td>
<td>103</td>
<td>Protocol Independent Multicast.</td>
</tr>
<tr>
<td>pptp</td>
<td>47</td>
<td>Point-to-Point Tunneling Protocol. Entering the pptp protocol literal is equivalent to entering the gre protocol literal.</td>
</tr>
<tr>
<td>snp</td>
<td>109</td>
<td>Sitara Networks Protocol.</td>
</tr>
<tr>
<td>tcp</td>
<td>6</td>
<td>Transmission Control Protocol, RFC 793.</td>
</tr>
<tr>
<td>udp</td>
<td>17</td>
<td>User Datagram Protocol, RFC 768.</td>
</tr>
</tbody>
</table>

Protocol numbers can be viewed online at the IANA website:
http://www.iana.org/assignments/protocol-numbers

TCP and UDP Ports

Table B-7 lists the literal values and port numbers; either can be entered in ASASM commands. See the following caveats:

- The ASASM uses port 1521 for SQL*Net. This is the default port used by Oracle for SQL*Net. This value, however, does not agree with IANA port assignments.
- The ASASM listens for RADIUS on ports 1645 and 1646. If your RADIUS server uses the standard ports 1812 and 1813, you can configure the ASASM to listen to those ports using the authentication-port and accounting-port commands.
To assign a port for DNS access, use the **domain** literal value, not **dns**. If you use **dns**, the ASASM assumes you meant to use the **dnsix** literal value.

Port numbers can be viewed online at the IANA website:

[http://www.iana.org/assignments/port-numbers](http://www.iana.org/assignments/port-numbers)

### Table B-7 Port Literal Values

<table>
<thead>
<tr>
<th>Literal</th>
<th>TCP or UDP?</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>aol</td>
<td>TCP</td>
<td>5190</td>
<td>America Online</td>
</tr>
<tr>
<td>bgp</td>
<td>TCP</td>
<td>179</td>
<td>Border Gateway Protocol, RFC 1163</td>
</tr>
<tr>
<td>biff</td>
<td>UDP</td>
<td>512</td>
<td>Used by mail system to notify users that new mail is received</td>
</tr>
<tr>
<td>bootpc</td>
<td>UDP</td>
<td>68</td>
<td>Bootstrap Protocol Client</td>
</tr>
<tr>
<td>bootps</td>
<td>UDP</td>
<td>67</td>
<td>Bootstrap Protocol Server</td>
</tr>
<tr>
<td>chargen</td>
<td>TCP</td>
<td>19</td>
<td>Character Generator</td>
</tr>
<tr>
<td>citrix-ica</td>
<td>TCP</td>
<td>1494</td>
<td>Citrix Independent Computing Architecture (ICA) protocol</td>
</tr>
<tr>
<td>cmd</td>
<td>TCP</td>
<td>514</td>
<td>Similar to exec except that cmd has automatic authentication</td>
</tr>
<tr>
<td>ctiqbe</td>
<td>TCP</td>
<td>2748</td>
<td>Computer Telephony Interface Quick Buffer Encoding</td>
</tr>
<tr>
<td>daytime</td>
<td>TCP</td>
<td>13</td>
<td>Day time, RFC 867</td>
</tr>
<tr>
<td>discard</td>
<td>TCP, UDP</td>
<td>9</td>
<td>Discard</td>
</tr>
<tr>
<td>domain</td>
<td>TCP, UDP</td>
<td>53</td>
<td>DNS</td>
</tr>
<tr>
<td>dnsix</td>
<td>UDP</td>
<td>195</td>
<td>DNSIX Session Management Module Audit Redirector</td>
</tr>
<tr>
<td>echo</td>
<td>TCP, UDP</td>
<td>7</td>
<td>Echo</td>
</tr>
<tr>
<td>exec</td>
<td>TCP</td>
<td>512</td>
<td>Remote process execution</td>
</tr>
<tr>
<td>finger</td>
<td>TCP</td>
<td>79</td>
<td>Finger</td>
</tr>
<tr>
<td>ftp</td>
<td>TCP</td>
<td>21</td>
<td>File Transfer Protocol (control port)</td>
</tr>
<tr>
<td>ftp-data</td>
<td>TCP</td>
<td>20</td>
<td>File Transfer Protocol (data port)</td>
</tr>
<tr>
<td>gopher</td>
<td>TCP</td>
<td>70</td>
<td>Gopher</td>
</tr>
<tr>
<td>https</td>
<td>TCP</td>
<td>443</td>
<td>HTTP over SSL</td>
</tr>
<tr>
<td>h323</td>
<td>TCP</td>
<td>1720</td>
<td>H.323 call signalling</td>
</tr>
<tr>
<td>hostname</td>
<td>TCP</td>
<td>101</td>
<td>NIC Host Name Server</td>
</tr>
<tr>
<td>ident</td>
<td>TCP</td>
<td>113</td>
<td>Ident authentication service</td>
</tr>
<tr>
<td>imap4</td>
<td>TCP</td>
<td>143</td>
<td>Internet Message Access Protocol, version 4</td>
</tr>
<tr>
<td>irc</td>
<td>TCP</td>
<td>194</td>
<td>Internet Relay Chat protocol</td>
</tr>
<tr>
<td>isakmp</td>
<td>UDP</td>
<td>500</td>
<td>Internet Security Association and Key Management Protocol</td>
</tr>
<tr>
<td>kerberos</td>
<td>TCP, UDP</td>
<td>750</td>
<td>Kerberos</td>
</tr>
<tr>
<td>Literal</td>
<td>TCP or UDP?</td>
<td>Value</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-------------</td>
<td>-------</td>
<td>----------------------------------------------------------</td>
</tr>
<tr>
<td>klogin</td>
<td>TCP</td>
<td>543</td>
<td>KLOGIN</td>
</tr>
<tr>
<td>kshell</td>
<td>TCP</td>
<td>544</td>
<td>Korn Shell</td>
</tr>
<tr>
<td>ldap</td>
<td>TCP</td>
<td>389</td>
<td>Lightweight Directory Access Protocol</td>
</tr>
<tr>
<td>ldaps</td>
<td>TCP</td>
<td>636</td>
<td>Lightweight Directory Access Protocol (SSL)</td>
</tr>
<tr>
<td>lpd</td>
<td>TCP</td>
<td>515</td>
<td>Line Printer Daemon - printer spooler</td>
</tr>
<tr>
<td>login</td>
<td>TCP</td>
<td>513</td>
<td>Remote login</td>
</tr>
<tr>
<td>lotusnotes</td>
<td>TCP</td>
<td>1352</td>
<td>IBM Lotus Notes</td>
</tr>
<tr>
<td>mobile-ip</td>
<td>UDP</td>
<td>434</td>
<td>MobileIP-Agent</td>
</tr>
<tr>
<td>nameserver</td>
<td>UDP</td>
<td>42</td>
<td>Host Name Server</td>
</tr>
<tr>
<td>netbios-ns</td>
<td>UDP</td>
<td>137</td>
<td>NetBIOS Name Service</td>
</tr>
<tr>
<td>netbios-dgm</td>
<td>UDP</td>
<td>138</td>
<td>NetBIOS Datagram Service</td>
</tr>
<tr>
<td>netbios-ssn</td>
<td>TCP</td>
<td>139</td>
<td>NetBIOS Session Service</td>
</tr>
<tr>
<td>nntp</td>
<td>TCP</td>
<td>119</td>
<td>Network News Transfer Protocol</td>
</tr>
<tr>
<td>ntp</td>
<td>UDP</td>
<td>123</td>
<td>Network Time Protocol</td>
</tr>
<tr>
<td>pcanywhere-status</td>
<td>UDP</td>
<td>5632</td>
<td>pcAnywhere status</td>
</tr>
<tr>
<td>pcanywhere-data</td>
<td>TCP</td>
<td>5631</td>
<td>pcAnywhere data</td>
</tr>
<tr>
<td>pim-auto-rp</td>
<td>TCP, UDP</td>
<td>496</td>
<td>Protocol Independent Multicast, reverse path flooding, dense mode</td>
</tr>
<tr>
<td>pop2</td>
<td>TCP</td>
<td>109</td>
<td>Post Office Protocol - Version 2</td>
</tr>
<tr>
<td>pop3</td>
<td>TCP</td>
<td>110</td>
<td>Post Office Protocol - Version 3</td>
</tr>
<tr>
<td>pptp</td>
<td>TCP</td>
<td>1723</td>
<td>Point-to-Point Tunneling Protocol</td>
</tr>
<tr>
<td>radius</td>
<td>UDP</td>
<td>1645</td>
<td>Remote Authentication Dial-In User Service</td>
</tr>
<tr>
<td>radius-acct</td>
<td>UDP</td>
<td>1646</td>
<td>Remote Authentication Dial-In User Service (accounting)</td>
</tr>
<tr>
<td>rip</td>
<td>UDP</td>
<td>520</td>
<td>Routing Information Protocol</td>
</tr>
<tr>
<td>secureid-udp</td>
<td>UDP</td>
<td>5510</td>
<td>SecureID over UDP</td>
</tr>
<tr>
<td>smtp</td>
<td>TCP</td>
<td>25</td>
<td>Simple Mail Transport Protocol</td>
</tr>
<tr>
<td>snmp</td>
<td>UDP</td>
<td>161</td>
<td>Simple Network Management Protocol</td>
</tr>
<tr>
<td>snmptrap</td>
<td>UDP</td>
<td>162</td>
<td>Simple Network Management Protocol - Trap</td>
</tr>
<tr>
<td>sqlnet</td>
<td>TCP</td>
<td>1521</td>
<td>Structured Query Language Network</td>
</tr>
<tr>
<td>ssh</td>
<td>TCP</td>
<td>22</td>
<td>Secure Shell</td>
</tr>
<tr>
<td>sunrpc (rpc)</td>
<td>TCP, UDP</td>
<td>111</td>
<td>Sun Remote Procedure Call</td>
</tr>
<tr>
<td>syslog</td>
<td>UDP</td>
<td>514</td>
<td>System Log</td>
</tr>
<tr>
<td>tacacs</td>
<td>TCP, UDP</td>
<td>49</td>
<td>Terminal Access Controller Access Control System Plus</td>
</tr>
<tr>
<td>talk</td>
<td>TCP, UDP</td>
<td>517</td>
<td>Talk</td>
</tr>
<tr>
<td>telnet</td>
<td>TCP</td>
<td>23</td>
<td>RFC 854 Telnet</td>
</tr>
</tbody>
</table>
Local Ports and Protocols

Table B-7 lists the protocols, TCP ports, and UDP ports that the ASASM may open to process traffic destined to the ASASM. Unless you enable the features and services listed in Table B-8, the ASASM does not open any local protocols or any TCP or UDP ports. You must configure a feature or service for the ASASM to open the default listening protocol or port. In many cases you can configure ports other than the default port when you enable a feature or service.

### Table B-7 Port Literal Values (continued)

<table>
<thead>
<tr>
<th>Literal</th>
<th>TCP or UDP?</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>tftp</td>
<td>UDP</td>
<td>69</td>
<td>Trivial File Transfer Protocol</td>
</tr>
<tr>
<td>time</td>
<td>UDP</td>
<td>37</td>
<td>Time</td>
</tr>
<tr>
<td>uucp</td>
<td>TCP</td>
<td>540</td>
<td>UNIX-to-UNIX Copy Program</td>
</tr>
<tr>
<td>who</td>
<td>UDP</td>
<td>513</td>
<td>Who</td>
</tr>
<tr>
<td>whois</td>
<td>TCP</td>
<td>43</td>
<td>Who Is</td>
</tr>
<tr>
<td>www</td>
<td>TCP</td>
<td>80</td>
<td>World Wide Web</td>
</tr>
<tr>
<td>xdmcp</td>
<td>UDP</td>
<td>177</td>
<td>X Display Manager Control Protocol</td>
</tr>
</tbody>
</table>

### Table B-8 Protocols and Ports Opened by Features and Services

<table>
<thead>
<tr>
<th>Feature or Service</th>
<th>Protocol</th>
<th>Port Number</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>DHCP</td>
<td>UDP</td>
<td>67,68</td>
<td></td>
</tr>
<tr>
<td>Failover Control</td>
<td>UDP</td>
<td>105</td>
<td>N/A</td>
</tr>
<tr>
<td>HTTP</td>
<td>TCP</td>
<td>80</td>
<td></td>
</tr>
<tr>
<td>HTTPS</td>
<td>TCP</td>
<td>443</td>
<td></td>
</tr>
<tr>
<td>ICMP</td>
<td>TCP</td>
<td>1</td>
<td>N/A</td>
</tr>
<tr>
<td>IGMP</td>
<td>TCP</td>
<td>2</td>
<td>Protocol only open on destination IP address 224.0.0.1</td>
</tr>
<tr>
<td>ISAKMP/IKE</td>
<td>UDP</td>
<td>500</td>
<td>Configurable.</td>
</tr>
<tr>
<td>IPsec (ESP)</td>
<td>UDP</td>
<td>50</td>
<td>N/A</td>
</tr>
<tr>
<td>IPsec over UDP (NAT-T)</td>
<td>UDP</td>
<td>4500</td>
<td></td>
</tr>
<tr>
<td>IPsec over UDP (Cisco VPN 3000 Series compatible)</td>
<td>UDP</td>
<td>10000</td>
<td>Configurable.</td>
</tr>
<tr>
<td>IPsec over TCP (CTCP)</td>
<td>TCP</td>
<td></td>
<td>No default port is used. You must specify the port number when configuring IPsec over TCP.</td>
</tr>
<tr>
<td>NTP</td>
<td>UDP</td>
<td>123</td>
<td></td>
</tr>
<tr>
<td>OSPF</td>
<td>TCP</td>
<td>89</td>
<td>Protocol only open on destination IP address 224.0.0.5 and 224.0.0.6</td>
</tr>
</tbody>
</table>
ICMP Types

Table B-9 lists the ICMP type numbers and names that you can enter in ASASM commands.

<table>
<thead>
<tr>
<th>ICMP Number</th>
<th>ICMP Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>echo-reply</td>
</tr>
<tr>
<td>3</td>
<td>unreachable</td>
</tr>
<tr>
<td>4</td>
<td>source-quench</td>
</tr>
<tr>
<td>5</td>
<td>redirect</td>
</tr>
<tr>
<td>6</td>
<td>alternate-address</td>
</tr>
<tr>
<td>8</td>
<td>echo</td>
</tr>
<tr>
<td>9</td>
<td>router-advertisement</td>
</tr>
<tr>
<td>10</td>
<td>router-solicitation</td>
</tr>
<tr>
<td>11</td>
<td>time-exceeded</td>
</tr>
<tr>
<td>12</td>
<td>parameter-problem</td>
</tr>
<tr>
<td>13</td>
<td>timestamp-request</td>
</tr>
<tr>
<td>14</td>
<td>timestamp-reply</td>
</tr>
<tr>
<td>15</td>
<td>information-request</td>
</tr>
<tr>
<td>16</td>
<td>information-reply</td>
</tr>
<tr>
<td>17</td>
<td>mask-request</td>
</tr>
<tr>
<td>18</td>
<td>mask-reply</td>
</tr>
<tr>
<td>31</td>
<td>conversion-error</td>
</tr>
<tr>
<td>32</td>
<td>mobile-redirect</td>
</tr>
</tbody>
</table>

Table B-8  Protocols and Ports Opened by Features and Services (continued)

<table>
<thead>
<tr>
<th>Feature or Service</th>
<th>Protocol</th>
<th>Port Number</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>PIM</td>
<td>103</td>
<td>N/A</td>
<td>Protocol only open on destination IP address 224.0.0.13</td>
</tr>
<tr>
<td>RIP</td>
<td>UDP</td>
<td>520</td>
<td>—</td>
</tr>
<tr>
<td>RIPv2</td>
<td>UDP</td>
<td>520</td>
<td>Port only open on destination IP address 224.0.0.9</td>
</tr>
<tr>
<td>SNMP</td>
<td>UDP</td>
<td>161</td>
<td>Configurable.</td>
</tr>
<tr>
<td>SSH</td>
<td>TCP</td>
<td>22</td>
<td>—</td>
</tr>
<tr>
<td>Stateful Update</td>
<td>8 and 9</td>
<td>N/A</td>
<td>—</td>
</tr>
<tr>
<td>Telnet</td>
<td>TCP</td>
<td>23</td>
<td>—</td>
</tr>
<tr>
<td>VPN Load Balancing</td>
<td>UDP</td>
<td>9023</td>
<td>Configurable.</td>
</tr>
<tr>
<td>VPN Individual User Authentication Proxy</td>
<td>UDP</td>
<td>1645, 1646</td>
<td>Port accessible only over VPN tunnel.</td>
</tr>
</tbody>
</table>
Configuring an External Server for Authorization and Authentication

This appendix describes how to configure an external LDAP, RADIUS, or TACACS+ server to support AAA on the ASASM. Before you configure the ASASM to use an external server, you must configure the server with the correct ASASM authorization attributes and, from a subset of these attributes, assign specific permissions to individual users.

This appendix includes the following sections:

- Understanding Policy Enforcement of Permissions and Attributes, page C-1
- Configuring an External LDAP Server, page C-2
- Configuring an External RADIUS Server, page C-27
- RADIUS Accounting Disconnect Reason Codes, page C-37

Understanding Policy Enforcement of Permissions and Attributes

The ASASM supports several methods of applying user authorization attributes (also called user entitlements or permissions) to VPN connections. You can configure the ASASM to obtain user attributes from a Dynamic Access Policy (DAP) on the ASASM, from an external authentication and/or authorization AAA server (RADIUS or LDAP), from a group policy on the ASASM, or from all three.

If the ASASM receives attributes from all sources, the attributes are evaluated, merged, and applied to the user policy. If there are conflicts between attributes coming from the DAP, the AAA server, or the group policy, those attributes obtained from the DAP always take precedence.

The ASASM applies attributes in the following order (see Figure C-1):

1. DAP attributes on the ASASM—Introduced in Version 8.0(2), these attributes take precedence over all others. If you set a bookmark or URL list in DAP, it overrides a bookmark or URL list set in the group policy.
2. User attributes on the AAA server—The server returns these attributes after successful user authentication and/or authorization. Do not confuse these with attributes that are set for individual users in the local AAA database on the ASASM (User Accounts in ASDM).
3. Group policy configured on the ASASM—If a RADIUS server returns the value of the RADIUS CLASS attribute IETF-Class-25 (OU=group-policy) for the user, the ASASM places the user in the group policy of the same name and enforces any attributes in the group policy that are not returned by the server.

For LDAP servers, any attribute name can be used to set the group policy for the session. The LDAP attribute map that you configure on the ASASM maps the LDAP attribute to the Cisco attribute IETF-Radius-Class.

4. Group policy assigned by the Connection Profile (called tunnel-group in the CLI)—The Connection Profile has the preliminary settings for the connection, and includes a default group policy applied to the user before authentication. All users connecting to the ASASM initially belong to this group, which provides any attributes that are missing from the DAP, user attributes returned by the server, or the group policy assigned to the user.

5. Default group policy assigned by the ASASM (DfltGrpPolicy)—System default attributes provide any values that are missing from the DAP, user attributes, group policy, or connection profile.

Figure C-1  Policy Enforcement Flow

Configuring an External LDAP Server

The VPN 3000 concentrator and the ASA/PIX 7.0 software required a Cisco LDAP schema for authorization operations. Beginning with Version 7.1.x, the ASASM performs authentication and authorization using the native LDAP schema, and the Cisco schema is no longer needed.

You configure authorization (permission policy) using an LDAP attribute map. For examples, see the “Active Directory/LDAP VPN Remote Access Authorization Examples” section on page C-16.

This section describes the structure, schema, and attributes of an LDAP server and includes the following topics:

- Organizing the ASASM for LDAP Operations, page C-3
- Defining the ASASM LDAP Configuration, page C-5
Note

For more information about the LDAP protocol, see RFCs 1777, 2251, and 2849.

Organizing the ASASM for LDAP Operations

This section describes how to search within the LDAP hierarchy and perform authenticated binding to the LDAP server on the ASASM and includes the following topics:

- Searching the LDAP Hierarchy, page C-3
- Binding the ASASM to the LDAP Server, page C-4

Your LDAP configuration should reflect the logical hierarchy of your organization. For example, suppose an employee at your company, Example Corporation, is named Employee1. Employee1 works in the Engineering group. Your LDAP hierarchy could have one or many levels. You might decide to set up a single-level hierarchy in which Employee1 is considered a member of Example Corporation. Or you could set up a multi-level hierarchy in which Employee1 is considered to be a member of the department Engineering, which is a member of an organizational unit called People, which is itself a member of Example Corporation. See Figure C-2 for an example of a multi-level hierarchy.

A multi-level hierarchy has more detail, but searches return results more quickly in a single-level hierarchy.

Figure C-2  A Multi-Level LDAP Hierarchy

Searching the LDAP Hierarchy

The ASASM lets you tailor the search within the LDAP hierarchy. You configure the following three fields on the ASASM to define where in the LDAP hierarchy that your search begins, the extent, and the type of information it is looking for. Together these fields allow you to limit the search of the hierarchy to only the part that includes the user permissions.

- LDAP Base DN defines where in the LDAP hierarchy that the server should begin searching for user information when it receives an authorization request from the ASASM.
Appendix C  Configuring an External Server for Authorization and Authentication

Configuring an External LDAP Server

Search Scope defines the extent of the search in the LDAP hierarchy. The search proceeds this many levels in the hierarchy below the LDAP Base DN. You can choose to have the server search only the level immediately below it, or it can search the entire subtree. A single level search is quicker, but a subtree search is more extensive.

Naming Attribute(s) defines the RDN that uniquely identifies an entry in the LDAP server. Common naming attributes can include cn (Common Name), sAMAccountName, and userPrincipalName.

Figure C-2 shows a sample LDAP hierarchy for Example Corporation. Given this hierarchy, you could define your search in different ways. Table C-1 shows two sample search configurations.

In the first example configuration, when Employee1 establishes the IPsec tunnel with LDAP authorization required, the ASASM sends a search request to the LDAP server, indicating it should search for Employee1 in the Engineering group. This search is quick.

In the second example configuration, the ASASM sends a search request indicating that the server should search for Employee1 within Example Corporation. This search takes longer.

Table C-1  Example Search Configurations

<table>
<thead>
<tr>
<th>No.</th>
<th>LDAP Base DN</th>
<th>Search Scope</th>
<th>Naming Attribute</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>group= Engineering,ou=People,dc=ExampleCorporation, dc=com</td>
<td>One Level</td>
<td>cn=Employee1</td>
<td>Quicker search</td>
</tr>
<tr>
<td>2</td>
<td>dc=ExampleCorporation,dc=com</td>
<td>Subtree</td>
<td>cn=Employee1</td>
<td>Longer search</td>
</tr>
</tbody>
</table>

Binding the ASASM to the LDAP Server

Some LDAP servers (including the Microsoft Active Directory server) require the ASASM to establish a handshake via authenticated binding before they accept requests for any other LDAP operations. The ASASM uses the Login Distinguished Name (DN) and Login Password to establish a trust relationship (bind) with an LDAP server before a user can search. The Login DN represents a user record in the LDAP server that the administrator uses for binding.

When binding, the ASASM authenticates to the server using the Login DN and the Login Password. When performing a Microsoft Active Directory read-only operation (such as for authentication, authorization, or group search), the ASASM can bind with a Login DN with fewer privileges. For example, the Login DN can be a user whose AD “Member Of” designation is part of Domain Users. For VPN password management write operations, the Login DN needs elevated privileges and must be part of the Account Operators AD group. Microsoft Active Directory group search (also called “MemberOf retrieval”) was added in ASASM Version 8.0.4.

An example of a Login DN includes the following entries:
cn=Binduser1,ou=Admins,ou= Users,dc=company_A,dc=com

See your LDAP Administrator guide for specific Login DN requirements for read and write operations.

The ASASM supports the following features:

- Simple LDAP authentication with an unencrypted password using the default port 389. You can also use other ports instead of the default port.
- Secure LDAP (LDAP-S) using the default port 636. You can also use other ports instead of the default port.
- Simple Authentication and Security Layer (SASL) MD5
- SASL Kerberos

The ASASM does not support anonymous authentication.
As an LDAP client, the ASASM does not support the transmission of anonymous binds or requests.

**Defining the ASASM LDAP Configuration**

This section describes how to define the LDAP AV-pair attribute syntax and includes the following topics:

- Supported Cisco Attributes for LDAP Authorization, page C-5
- Cisco AV Pair Attribute Syntax, page C-13
- Cisco AV Pairs ACL Examples, page C-14

The ASASM enforces the LDAP attributes based on attribute name, not numeric ID. RADIUS attributes, on the other hand, are enforced by numeric ID, not by name.

Authorization refers to the process of enforcing permissions or attributes. An LDAP server defined as an authentication or authorization server enforces permissions or attributes if they are configured.

For software Version 7.0, LDAP attributes include the cVPN3000 prefix. For software Versions 7.1 and later, this prefix was removed.

**Supported Cisco Attributes for LDAP Authorization**

This section provides a complete list of attributes (see Table C-2) for the ASA 5500, VPN 3000 concentrator, and PIX 500 series ASASMs. The table includes attribute support information for the VPN 3000 concentrator and PIX 500 series ASASMs to assist you in configuring networks with a combination of these devices.
### Table C-2  ASASM Supported Cisco Attributes for LDAP Authorization

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>VPN 3000</th>
<th>ASA</th>
<th>PIX</th>
<th>Syntax/Type</th>
<th>Single or Multi-Valued</th>
<th>Possible Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access-Hours</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>String</td>
<td>Single</td>
<td>Name of the time-range (for example, Business-Hours)</td>
</tr>
<tr>
<td>Allow-Network-Extension- Mode</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Boolean</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>Authenticated-User-Idle- Timeout</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>1 - 35791394 minutes</td>
</tr>
<tr>
<td>Authorization-Required</td>
<td>Y</td>
<td></td>
<td></td>
<td>Integer</td>
<td>Single</td>
<td>0 = No 1 = Yes</td>
</tr>
<tr>
<td>Authorization-Type</td>
<td>Y</td>
<td></td>
<td></td>
<td>Integer</td>
<td>Single</td>
<td>0 = None 1 = RADIUS 2 = LDAP</td>
</tr>
<tr>
<td>Banner1</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>String</td>
<td>Single</td>
<td>Banner string for clientless and client SSL VPN, and IPsec clients.</td>
</tr>
<tr>
<td>Banner2</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>String</td>
<td>Single</td>
<td>Banner string for clientless and client SSL VPN, and IPsec clients.</td>
</tr>
<tr>
<td>Cisco-AV-Pair</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>String</td>
<td>Multi</td>
<td>An octet string in the following format:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>[Prefix] [Action] [Protocol] [Source] [Source Wildcard Mask] [Destination] [Destination Wildcard Mask] [Established] [Log] [Operator] [Port]</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>For more information, see the “Cisco AV Pair Attribute Syntax” section on page C-13.”</td>
</tr>
<tr>
<td>Cisco-IP-Phone-Bypass</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>Cisco-LEAP-Bypass</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>Client-Intercept-DHCP-Configure-Msg</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Boolean</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>Client-Type-Version-Limiting</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>String</td>
<td>Single</td>
<td>IPsec VPN client version number string</td>
</tr>
<tr>
<td>Confidence-Interval</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>10 - 300 seconds</td>
</tr>
<tr>
<td>DHCP-Network-Scope</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>String</td>
<td>Single</td>
<td>IP address</td>
</tr>
<tr>
<td>Firewall-ACL-In</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>String</td>
<td>Single</td>
<td>Access list ID</td>
</tr>
<tr>
<td>Firewall-ACL-Out</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>String</td>
<td>Single</td>
<td>Access list ID</td>
</tr>
</tbody>
</table>
### Table C-2  ASASM Supported Cisco Attributes for LDAP Authorization (continued)

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>VPN 3000</th>
<th>ASA</th>
<th>PIX</th>
<th>Syntax/Type</th>
<th>Single or Multi-Valued</th>
<th>Possible Values</th>
</tr>
</thead>
</table>
| Group-Policy            | Y        | Y   | Y   | String      | Single                 | Sets the group policy for the remote access VPN session. For version 8.2 and later, use this attribute instead of IETF-Radius-Class. You can use one of the three following formats:  
- group policy name  
- OU=group policy name  
- OU=group policy name: |
| IE-Proxy-Bypass-Local   |          |     |     | Boolean     | Single                 | 0=Disabled  
1=Enabled |
| IE-Proxy-Exception-List |          |     |     | String      | Single                 | A list of DNS domains. Entries must be separated by the newline character sequence (\n). |
| IE-Proxy-Method         | Y        | Y   | Y   | Integer     | Single                 | 1 = Do not modify proxy settings  
2 = Do not use proxy  
3 = Auto detect  
4 = Use ASASM setting |
| IE-Proxy-Server         | Y        | Y   | Y   | Integer     | Single                 | IP address |
| IETF-Radius-Class       | Y        | Y   | Y   | String      | Single                 | Sets the group policy for the remote access VPN session. For versions 8.2 and later, we recommend that you use the Group-Policy attribute. You can use one of the three following formats:  
- group policy name  
- OU=group policy name  
- OU=group policy name: |
| IETF-Radius-Filter-Id   | Y        | Y   | Y   | String      | Single                 | Access list name that is defined on the ASASM. The setting applies to VPN remote access IPsec and SSL VPN clients. |
| IETF-Radius-Framed-IP-Address | Y        | Y   | Y   | String      | Single                 | An IP address. The setting applies to VPN remote access IPsec and SSL VPN clients. |
| IETF-Radius-Framed-IP-Netmask | Y        | Y   | Y   | String      | Single                 | An IP address mask. The setting applies to VPN remote access IPsec and SSL VPN clients. |
| IETF-Radius-Idle-Timeout| Y        | Y   | Y   | Integer     | Single                 | Seconds |
### Table C-2: ASASM Supported Cisco Attributes for LDAP Authorization (continued)

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>VPN 3000</th>
<th>ASA</th>
<th>PIX</th>
<th>Syntax/Type</th>
<th>Single or Multi-Valued</th>
<th>Possible Values</th>
</tr>
</thead>
</table>
| IETF-Radius-Service-Type              | Y         | Y   | Y   | Integer     | Single                 | 1 = Login  
2 = Framed  
5 = Remote access  
6 = Administrative  
7 = NAS prompt                                                   |
| IETF-Radius-Session-Timeout           | Y         | Y   | Y   | Integer     | Single                 | Seconds                                                                                                                                   |
| IKE-Keep-Alives                       | Y         | Y   | Y   | Boolean     | Single                 | 0 = Disabled  
1 = Enabled                                                                                                                              |
| IPsec-Allow-Passwd-Store              | Y         | Y   | Y   | Boolean     | Single                 | 0 = Disabled  
1 = Enabled                                                                                                                              |
| IPsec-Authentication                  | Y         | Y   | Y   | Integer     | Single                 | 0 = None  
1 = RADIUS  
2 = LDAP (authorization only)  
3 = NT Domain  
4 = SDI (RSA)  
5 = Internal  
6 = RADIUS with Expiry  
7 = Kerberos or Active Directory                                             |
| IPsec-Auth-On-Rekey                   | Y         | Y   | Y   | Boolean     | Single                 | 0 = Disabled  
1 = Enabled                                                                                                                              |
| IPsec-Backup-Server-List              | Y         | Y   | Y   | String      | Single                 | Server addresses (space delimited)                                                                                                          |
| IPsec-Backup-Servers                  | Y         | Y   | Y   | String      | Single                 | 1 = Use client-configured list  
2 = Disabled and clear client list  
3 = Use backup server list                                                                                                                  |
| IPsec-Client-Firewall-Filter-Name     | Y         |     |     | String      | Single                 | Specifies the name of the filter to be pushed to the client as firewall policy.                                                           |
| IPsec-Client-Firewall-Filter-Optional | Y         | Y   | Y   | Integer     | Single                 | 0 = Required  
1 = Optional                                                                                                                             |
| IPsec-Default-Domain                  | Y         | Y   | Y   | String      | Single                 | Specifies the single default domain name to send to the client (1 - 255 characters).                                                   |
| IPsec-Extended-Auth-On-Rekey          |           | Y   | Y   | String      | Single                 | String                                                                                                                                     |
| IPsec-IKE-Peer-ID-Check               | Y         | Y   | Y   | Integer     | Single                 | 1 = Required  
2 = If supported by peer certificate  
3 = Do not check                                                                                                                           |
| IPsec-IP-Compression                  | Y         | Y   | Y   | Integer     | Single                 | 0 = Disabled  
1 = Enabled                                                                                                                              |
| IPsec-Mode-Config                     | Y         | Y   | Y   | Boolean     | Single                 | 0 = Disabled  
1 = Enabled                                                                                                                              |
| IPsec-Over-UDP                        | Y         | Y   | Y   | Boolean     | Single                 | 0 = Disabled  
1 = Enabled                                                                                                                              |
### Table C-2: ASASM Supported Cisco Attributes for LDAP Authorization (continued)

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>VPN 3000</th>
<th>ASA</th>
<th>PIX</th>
<th>Syntax/Type</th>
<th>Single or Multi-Valued</th>
<th>Possible Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPsec-Over-UDP-Port</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>4001 - 49151; The default is 10000.</td>
</tr>
<tr>
<td>IPsec-Required-Client-Firewall-Capability</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>0 = None 1 = Policy defined by remote FW Are-You-There (AYT) 2 = Policy pushed CPP 4 = Policy from server</td>
</tr>
<tr>
<td>IPsec-Sec-Association</td>
<td>Y</td>
<td></td>
<td></td>
<td>String</td>
<td>Single</td>
<td>Name of the security association</td>
</tr>
<tr>
<td>IPsec-Split-DNS-Names</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>String</td>
<td>Single</td>
<td>Specifies the list of secondary domain names to send to the client (1 - 255 characters).</td>
</tr>
<tr>
<td>IPsec-Split-Tunneling-Policy</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Tunnel everything 1 = Split tunneling 2 = Local LAN permitted</td>
</tr>
<tr>
<td>IPsec-Split-Tunnel-List</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>String</td>
<td>Single</td>
<td>Specifies the name of the network or access list that describes the split tunnel inclusion list.</td>
</tr>
<tr>
<td>IPsec-Tunnel-Type</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>1 = LAN-to-LAN 2 = Remote access</td>
</tr>
<tr>
<td>IPsec-User-Group-Lock</td>
<td>Y</td>
<td></td>
<td></td>
<td>Boolean</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>L2TP-MPPC-Compression</td>
<td>Y</td>
<td></td>
<td></td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>MS-Client-Subnet-Mask</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>String</td>
<td>Single</td>
<td>An IP address</td>
</tr>
<tr>
<td>PFS-Required</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Boolean</td>
<td>Single</td>
<td>0 = No 1 = Yes</td>
</tr>
<tr>
<td>Port-Forwarding-Name</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>String</td>
<td>Single</td>
<td>Name string (for example, &quot;Corporate-Apps&quot;)</td>
</tr>
<tr>
<td>PPTP-MPPC-Compression</td>
<td>Y</td>
<td></td>
<td></td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
</tbody>
</table>
### Table C-2 ASASM Supported Cisco Attributes for LDAP Authorization (continued)

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>VPN 3000</th>
<th>ASA</th>
<th>PIX</th>
<th>Syntax/Type</th>
<th>Single or Multi-Valued</th>
<th>Possible Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Primary-DNS</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>String</td>
<td>Single</td>
<td>An IP address</td>
</tr>
<tr>
<td>Primary-WINS</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>String</td>
<td>Single</td>
<td>An IP address</td>
</tr>
<tr>
<td>Privilege-Level</td>
<td></td>
<td></td>
<td></td>
<td>Integer</td>
<td>Single</td>
<td>For usernames, 0 - 15</td>
</tr>
<tr>
<td>Required-Client-Firewall-Vendor-Code</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>1 = Cisco Systems (with Cisco Integrated Client)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2 = Zone Labs</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3 = NetworkICE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4 = Sygate</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>5 = Cisco Systems (with Cisco Intrusion Prevention Security Agent)</td>
</tr>
<tr>
<td>Required-Client-Firewall-Description</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>String</td>
<td>Single</td>
<td>—</td>
</tr>
<tr>
<td>Required-Client-Firewall-Product-Code</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>Cisco Systems Products:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1 = Cisco Intrusion Prevention Security Agent or Cisco Integrated Client (CIC)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Zone Labs Products:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1 = Zone Alarm</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2 = Zone AlarmPro</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3 = Zone Labs Integrity</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>NetworkICE Product:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1 = BlackICE Defender/Agent</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Sygate Products:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1 = Personal Firewall</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2 = Personal Firewall Pro</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3 = Security Agent</td>
</tr>
<tr>
<td>Require-HW-Client-Auth</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Boolean</td>
<td>Single</td>
<td>0 = Disabled</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1 = Enabled</td>
</tr>
<tr>
<td>Require-Individual-User-Auth</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1 = Enabled</td>
</tr>
<tr>
<td>Secondary-DNS</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>String</td>
<td>Single</td>
<td>An IP address</td>
</tr>
<tr>
<td>Secondary-WINS</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>String</td>
<td>Single</td>
<td>An IP address</td>
</tr>
<tr>
<td>SEP-Card-Assignment</td>
<td></td>
<td></td>
<td></td>
<td>Integer</td>
<td>Single</td>
<td>Not used</td>
</tr>
<tr>
<td>Simultaneous-Logins</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>0 - 2147483647</td>
</tr>
<tr>
<td>Strip-Realm</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Boolean</td>
<td>Single</td>
<td>0 = Disabled</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1 = Enabled</td>
</tr>
<tr>
<td>TACACS-Auth-type</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>—</td>
</tr>
<tr>
<td>TACACS-Privilege-Level</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>—</td>
</tr>
<tr>
<td>Tunnel-Group-Lock</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>String</td>
<td>Single</td>
<td>Name of the tunnel group or “none”</td>
</tr>
</tbody>
</table>
## Table C-2  ASASM Supported Cisco Attributes for LDAP Authorization (continued)

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>VPN 3000</th>
<th>ASA</th>
<th>PIX</th>
<th>Syntax/Type</th>
<th>Single or Multi-Valued</th>
<th>Possible Values</th>
</tr>
</thead>
</table>
| Tunneling-Protocols                | Y        | Y   | Y   | Integer     | Single                 | 1 = PPTP  
2 = L2TP  
4 = IPSec (IKEv1)  
8 = L2TP/IPSec  
16 = WebVPN  
32 = SVC  
64 = IPsec (IKEv2)  
8 and 4 are mutually exclusive  
(0 - 11, 16 - 27, 32 - 43, 48 - 59 are legal values). |
| Use-Client-Address                 | Y        |     |     | Boolean     | Single                 | 0 = Disabled  
1 = Enabled |
| User-Auth-Server-Name              | Y        |     |     | String      | Single                 | IP address or hostname |
| User-Auth-Server-Port              | Y        |     |     | Integer     | Single                 | Port number for server protocol |
| User-Auth-Server-Secret            | Y        |     |     | String      | Single                 | Server password |
| WebVPN-ACL-Filters                 |          |     | Y   | String      | Single                 | Webtype access list name |
| WebVPN-Apply-ACL-Enable            | Y        | Y   |     | Integer     | Single                 | 0 = Disabled  
1 = Enabled  
With Version 8.0 and later, this attribute is not required. |
| WebVPN-Citrix-Support-Enable       | Y        | Y   |     | Integer     | Single                 | 0 = Disabled  
1 = Enabled  
With Versions 8.0 and later, this attribute is not required. |
| WebVPN-Enable-functions            |          |     |     | Integer     | Single                 | Not used - deprecated |
| WebVPN-Exchange-Server-Address     |          |     |     | String      | Single                 | Not used - deprecated |
| WebVPN-Exchange-Server-NETBIOS-Name|          |     |     | String      | Single                 | Not used - deprecated |
| WebVPN-File-Access-Enable          | Y        | Y   |     | Integer     | Single                 | 0 = Disabled  
1 = Enabled |
| WebVPN-File-Server-Browsing-Enable | Y        | Y   |     | Integer     | Single                 | 0 = Disabled  
1 = Enabled |
| WebVPN-File-Server-Entry-Enable    | Y        | Y   |     | Integer     | Single                 | 0 = Disabled  
1 = Enabled |
| WebVPN-Forwarded-Ports             |          |     | Y   | String      | Single                 | Port-forward list name |
| WebVPN-Homepage                    | Y        | Y   |     | String      | Single                 | A URL such as http://www.example.com |
Table C-2 ASASM Supported Cisco Attributes for LDAP Authorization (continued)

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>VPN 3000</th>
<th>ASA</th>
<th>PIX</th>
<th>Syntax/Type</th>
<th>Single or Multi-Valued</th>
<th>Possible Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>WebVPN-Port-Forwarding-Auto-Download-Enable</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>WebVPN-Port-Forwarding-Enable</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>WebVPN-Port-Forwarding-Exchange-Proxy-Enable</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>WebVPN-Port-Forwarding-HTTP-Proxy-Enable</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>WebVPN-Single-Sign-On-Server-Name</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>String</td>
<td>Single</td>
<td>Name of the SSO Server (1 - 31 characters).</td>
</tr>
<tr>
<td>WebVPN-SVC-Client-DPD</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled n = Dead peer detection value in seconds (30 - 3600)</td>
</tr>
<tr>
<td>WebVPN-SVC-Compression</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>0 = None 1 = Deflate compression</td>
</tr>
<tr>
<td>WebVPN-SVC-Enable</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>WebVPN-SVC-Gateway-DPD</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled n = Dead peer detection value in seconds (30 - 3600)</td>
</tr>
<tr>
<td>WebVPN-SVC-Keepalive</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled n = Keepalive value in seconds (15 - 600)</td>
</tr>
<tr>
<td>WebVPN-SVC-Keep-Enable</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>WebVPN-SVC-Rekey-Method</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>0 = None 1 = SSL 2 = New tunnel 3 = Any (sets to SSL)</td>
</tr>
</tbody>
</table>
Table C-2  ASASM Supported Cisco Attributes for LDAP Authorization (continued)

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>VPN 3000</th>
<th>ASA</th>
<th>PIX</th>
<th>Syntax/Type</th>
<th>Single or Multi-Valued</th>
<th>Possible Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>WebVPN-SVC-Rekey-Period</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled \n n = Retry period in minutes (4 - 10080)</td>
</tr>
<tr>
<td>WebVPN-SVC-Required-Enable</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled \n 1 = Enabled</td>
</tr>
<tr>
<td>WebVPN-URL-Entry-Enable</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled \n 1 = Enabled</td>
</tr>
<tr>
<td>WebVPN-URL-List</td>
<td></td>
<td>Y</td>
<td></td>
<td>String</td>
<td>Single</td>
<td>URL list name</td>
</tr>
</tbody>
</table>

**Cisco AV Pair Attribute Syntax**

The Cisco Attribute Value (AV) pair (ID Number 26/9/1) can be used to enforce access lists from a RADIUS server (like Cisco ACS), or from an LDAP server via an LDAP attribute map.

The syntax of each Cisco-AV-Pair rule is as follows:

[Prefix] [Action] [Protocol] [Source] [Source Wildcard Mask] [Destination] [Destination Wildcard Mask] [Established] [Log] [Operator] [Port]

Table C-3 describes the syntax rules.

**Table C-3  AV-Pair Attribute Syntax Rules**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action</td>
<td>Action to perform if the rule matches a deny or a permit.</td>
</tr>
<tr>
<td>Destination</td>
<td>Network or host that receives the packet. Specify it as an IP address, a hostname, or the any keyword. If using an IP address, the source wildcard mask must follow.</td>
</tr>
<tr>
<td>Destination Wildcard Mask</td>
<td>The wildcard mask that applies to the destination address.</td>
</tr>
<tr>
<td>Log</td>
<td>Generates a FILTER log message. You must use this keyword to generate events of severity level 9.</td>
</tr>
<tr>
<td>Operator</td>
<td>Logic operators: greater than, less than, equal to, not equal to.</td>
</tr>
<tr>
<td>Port</td>
<td>The number of a TCP or UDP port in the range of 0 - 65535.</td>
</tr>
<tr>
<td>Prefix</td>
<td>A unique identifier for the AV pair (for example: ip:inavl#1= for standard access lists or webvpn:inavl# = for clientless SSL VPN access lists). This field only appears when the filter has been sent as an AV pair.</td>
</tr>
<tr>
<td>Protocol</td>
<td>Number or name of an IP protocol. Either an integer in the range of 0 - 255 or one of the following keywords: icmp, igmp, ip, tcp, udp.</td>
</tr>
</tbody>
</table>
Cisco AV Pairs ACL Examples

Table C-4 shows examples of Cisco AV pairs and describes the permit or deny actions that result.

Each ACL # in inacl# must be unique. However, they do not need to be sequential (for example, 1, 2, 3, 4). That is, they could be 5, 45, 135.

### Table C-4 Examples of Cisco AV Pairs and Their Permitting or Denying Action

<table>
<thead>
<tr>
<th>Cisco AV Pair Example</th>
<th>Permitting or Denying Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip:inacl#1=deny ip 10.155.10.0 0.0.0.255</td>
<td>Allows IP traffic between the two hosts using a full tunnel IPsec or SSL VPN client.</td>
</tr>
<tr>
<td>ip:inacl#2=permit TCP any host 10.160.0.1 eq 80 log</td>
<td>Allows TCP traffic from all hosts to the specific host on port 80 only using a full tunnel IPsec or SSL VPN client.</td>
</tr>
<tr>
<td>webvpn:inacl#1=permit url <a href="http://www.example.com">http://www.example.com</a></td>
<td>Allows clientless SSL VPN traffic to the URL specified, denies SMTP traffic to a specific server, and allows file share access (CIFS) to the specified server.</td>
</tr>
<tr>
<td>webvpn:inacl#2=deny url smtp://server cifs://server/share</td>
<td>Denies Telnet access and permits SSH access on non-default ports 2323 and 2222, respectively, or other application traffic flows using these ports for clientless SSL VPN.</td>
</tr>
<tr>
<td>webvpn:inacl#1=permit tcp 10.86.1.2 eq 2222 log</td>
<td>Allows clientless SSL VPN SSH access to default port 22 and denies Telnet access to port 23, respectively. This example assumes that you are using Telnet or SSH Java plug-ins enforced by these ACLs.</td>
</tr>
</tbody>
</table>

URL Types Supported in ACLs

The URL may be a partial URL, contain wildcards for the server, or include a port.
The following URL types are supported.

<table>
<thead>
<tr>
<th>Any All URLs</th>
<th>https://</th>
<th>post://</th>
<th>ssh://</th>
</tr>
</thead>
<tbody>
<tr>
<td>citrixs://</td>
<td>ftp://</td>
<td>smart-tunnel://</td>
<td></td>
</tr>
<tr>
<td>http://</td>
<td>pop3://</td>
<td>smtp://</td>
<td></td>
</tr>
</tbody>
</table>

Note: The URLs listed in this table appear in CLI or ASDM menus based on whether or not the associated plug-in is enabled.

Guidelines for Using Cisco-AV Pairs (ACLs)

- Use Cisco-AV pair entries with the ip:inacl# prefix to enforce access lists for remote IPsec and SSL VPN Client (SVC) tunnels.
- Use Cisco-AV pair entries with the webvpn:inacl# prefix to enforce access lists for SSL VPN clientless (browser-mode) tunnels.
- For webtype ACLs, you do not specify the source because the ASASM is the source.

Table C-5 lists the tokens for the Cisco-AV-pair attribute:

<table>
<thead>
<tr>
<th>Table C-5 ASASM-Supported Tokens</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Token</strong></td>
</tr>
<tr>
<td>ip:inacl#Num=</td>
</tr>
<tr>
<td>webvpn:inacl#Num=</td>
</tr>
<tr>
<td>deny</td>
</tr>
<tr>
<td>permit</td>
</tr>
<tr>
<td>icmp</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>IP</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>TCP</td>
</tr>
<tr>
<td>6</td>
</tr>
<tr>
<td>UDP</td>
</tr>
<tr>
<td>17</td>
</tr>
<tr>
<td>any</td>
</tr>
<tr>
<td>host</td>
</tr>
<tr>
<td>log</td>
</tr>
<tr>
<td>lt</td>
</tr>
</tbody>
</table>
Active Directory/LDAP VPN Remote Access Authorization Examples

This section presents example procedures for configuring authentication and authorization on the ASASM using the Microsoft Active Directory server. It includes the following topics:

- User-Based Attributes Policy Enforcement, page C-16
- Placing LDAP Users in a Specific Group Policy, page C-18
- Enforcing Static IP Address Assignment for AnyConnect Tunnels, page C-20
- Enforcing Dial-in Allow or Deny Access, page C-22
- Enforcing Logon Hours and Time-of-Day Rules, page C-25

Other configuration examples available on Cisco.com include the following TechNotes.

- ASA/PIX: Mapping VPN Clients to VPN Group Policies Through LDAP Configuration Example at the following URL:

- PIX/ASA 8.0: Use LDAP Authentication to Assign a Group Policy at Login at the following URL:

User-Based Attributes Policy Enforcement

You can map any standard LDAP attribute to a well-known Vendor-Specific Attribute (VSA) as well as map one or more LDAP attribute(s) to one or more Cisco LDAP attributes.

The following example shows how to configure the ASASM to enforce a simple banner for a user configured on an AD LDAP server. On the server, use the Office field in the General tab to enter the banner text. This field uses the attribute named physicalDeliveryOfficeName. On the ASASM, create an attribute map that maps physicalDeliveryOfficeName to the Cisco attribute Banner1. During authentication, the ASASM retrieves the value of physicalDeliveryOfficeName from the server, maps the value to the Cisco attribute Banner1, and displays the banner to the user.

This example applies to any connection type, including the IPsec VPN client, AnyConnect SSL VPN client, or clientless SSL VPN. In the example, User1 connects through a clientless SSL VPN connection.

To configure the attributes for a user on the AD or LDAP Server, perform the following steps:

**Step 1**
Right-click a user.

The Properties dialog box appears (see Figure C-3).
Step 2 Click the **General** tab and enter banner text in the Office field, which uses the AD/LDAP attribute `physicalDeliveryOfficeName`.

*Figure C-3  LDAP User Configuration*

![LDAP User Configuration](image)

Step 3 Create an LDAP attribute map on the ASASM.

The following example creates the map Banner and maps the AD/LDAP attribute `physicalDeliveryOfficeName` to the Cisco attribute `Banner1`:

```bash
hostname(config)# ldap attribute-map Banner
hostname(config-ldap-attribute-map)# map-name physicalDeliveryOfficeName Banner1
```

Step 4 Associate the LDAP attribute map to the AAA server.

The following example enters the aaa server host configuration mode for the host 10.1.1.2 in the AAA server group MS_LDAP, and associates the attribute map Banner that you created in Step 3:

```bash
hostname(config)# aaa-server MS_LDAP host 10.1.1.2
hostname(config-aaa-server-host)# ldap-attribute-map Banner
```

Step 5 Test the banner enforcement.

The following example shows a clientless SSL connection and the banner enforced through the attribute map after the user authenticates (see *Figure C-4*).
Placing LDAP Users in a Specific Group Policy

The following example shows how to authenticate User1 on the AD LDAP server to a specific group policy on the ASASM. On the server, use the Department field of the Organization tab to enter the name of the group policy. Then create an attribute map and map Department to the Cisco attribute IETF-Radius-Class. During authentication, the ASASM retrieves the value of Department from the server, maps the value to the IETF-Radius-Class, and places User1 in the group policy.

This example applies to any connection type, including the IPsec VPN client, AnyConnect SSL VPN client, or clientless SSL VPN. In this example, User1 is connecting through a clientless SSL VPN connection.

To configure the attributes for the user on the AD LDAP server, perform the following steps:

**Step 1**
Right-click the user.

The Properties dialog box appears (see Figure C-5).

**Step 2**
Click the **Organization** tab and enter **Group-Policy-1** in the Department field.
Step 3 Define an attribute map for the LDAP configuration shown in Step 1. The following example shows how to map the AD attribute Department to the Cisco attribute IETF-Radius-Class.

```
hostname(config)# ldap attribute-map group_policy
hostname(config-ldap-attribute-map)# map-name Department IETF-Radius-Class
```

Step 4 Associate the LDAP attribute map to the AAA server. The following example enters the aaa server host configuration mode for the host 10.1.1.2 in the AAA server group MS_LDAP, and associates the attribute map group_policy that you created in Step 3:

```
hostname(config)# aaa-server MS_LDAP host 10.1.1.2
hostname(config-aaa-server-host)# ldap-attribute-map group_policy
```

Step 5 Add the new group-policy on the ASASM and configure the required policy attributes that will be assigned to the user. The following example creates Group-policy-1, the name entered in the Department field on the server:

```
hostname(config)# group-policy Group-policy-1 external server-group LDAP_demo
hostname(config-group-policy)#
```

Step 6 Establish the VPN connection as the user would, and verify that the session inherits the attributes from Group-Policy1 (and any other applicable attributes from the default group-policy).

Step 7 Monitor the communication between the ASASM and the server by enabling the `debug ldap 255` command from privileged EXEC mode. The following is sample output from this command, which has been edited to provide the key messages:

```
[29] Authentication successful for user1 to 10.1.1.2
[29] Retrieving user attributes from server 10.1.1.2
```
Enforcing Static IP Address Assignment for AnyConnect Tunnels

In this example, configure the AnyConnect client user Web1 to receive a static IP address. Then enter the address in the Assign Static IP Address field of the Dialin tab on the AD LDAP server. This field uses the msRADIUSFramedIPAddress attribute. Create an attribute map that maps this attribute to the Cisco attribute IETF-Radius-Framed-IP-Address.

During authentication, the ASASM retrieves the value of msRADIUSFramedIPAddress from the server, maps the value to the Cisco attribute IETF-Radius-Framed-IP-Address, and provides the static address to User1.

The following example applies to full-tunnel clients, including the IPsec client and the SSL VPN clients (AnyConnect client 2.x and the SSL VPN client).

To configure the user attributes on the AD/LDAP server, perform the following steps:

**Step 1** Right-click the username.

The Properties dialog box appears (see Figure C-6).

**Step 2** Click the **Dialin** tab, check the **Assign Static IP Address** check box, and enter an IP address of 10.1.1.2.

**Step 3** Create an attribute map for the LDAP configuration shown in Step 1.
The following example shows how to map the AD attribute msRADIUSFramedIPAddress used by the Static Address field to the Cisco attribute IETF-Radius-Framed-IP-Address:

```
hostname(config)# ldap attribute-map static_address
hostname(config-ldap-attribute-map)# map-name msRADIUSFramedIPAddress
   IETF-Radius-Framed-IP-Address
```

**Step 4**  
Associate the LDAP attribute map to the AAA server.

The following example enters the aaa server host configuration mode for the host 10.1.1.2, in the AAA server group MS_LDAP, and associates the attribute map static_address that you created in Step 3:

```
hostname(config)# aaa-server MS_LDAP host 10.1.1.2
hostname(config-aaa-server-host)# ldap-attribute-map static_address
```

**Step 5**  
Verify that the `vpn-address-assignment` command is configured to specify AAA by viewing this part of the configuration with the `show run all vpn-addr-assign` command:

```
hostname(config)# show run all vpn-addr-assign
vpn-addr-assign aaa   << Make sure this is configured >>
no vpn-addr-assign dhcp
vpn-addr-assign local
hostname(config)#
```

**Step 6**  
Establish a connection to the ASASM with the AnyConnect client. Observe the following:

- The banner is received in the same sequence as a clientless connection (see Figure C-7).
- The user receives the IP address configured on the server and mapped to the ASASM (see Figure C-8).

*Figure C-7  Verify the Banner for the AnyConnect Session*
Enforcing Dial-in Allow or Deny Access

The following example creates an LDAP attribute map that specifies the tunneling protocols allowed by the user. You map the allow access and deny access settings on the Dialin tab to the Cisco attribute Tunneling-Protocol, which supports the bitmap values shown in Table C-6:

<table>
<thead>
<tr>
<th>Value</th>
<th>Tunneling Protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>PPTP</td>
</tr>
<tr>
<td>2</td>
<td>L2TP</td>
</tr>
<tr>
<td>41</td>
<td>IPsec (IKEv1)</td>
</tr>
<tr>
<td>82</td>
<td>L2TP/IPsec</td>
</tr>
</tbody>
</table>

Table C-6 Bitmap Values for Cisco Tunneling-Protocol Attribute
Appendix C Configuring an External Server for Authorization and Authentication

Configuring an External LDAP Server

Table C-6 Bitmap Values for Cisco Tunneling-Protocol Attribute (continued)

<table>
<thead>
<tr>
<th>Value</th>
<th>Tunneling Protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>Clientless SSL</td>
</tr>
<tr>
<td>32</td>
<td>SSL client—AnyConnect or SSL VPN client</td>
</tr>
<tr>
<td>64</td>
<td>IPsec (IKEv2)</td>
</tr>
</tbody>
</table>

1. IPsec and L2TP over IPsec are not supported simultaneously. Therefore, the values 4 and 8 are mutually exclusive.
2. See note 1.

Use this attribute to create an Allow Access (TRUE) or a Deny Access (FALSE) condition for the protocols and enforce the method for which the user is allowed access.

For this simplified example, by mapping the tunnel protocol IPsec/IKEv1 (4), you can create an allow (true) condition for the Cisco VPN client. You also map WebVPN (16) and SVC/AC (32), which are mapped as a value of 48 (16+32) and create a deny (false) condition. This allows the user to connect to the ASASM using IPsec, but any attempt to connect using clientless SSL or the AnyConnect client is denied.

Another example of enforcing dial-in allow access or deny access is available in the Tech Note ASA/PIX: Mapping VPN Clients to VPN Group Policies Through LDAP Configuration Example at the following URL:


To configure the user attributes on the AD/LDAP server, perform the following steps:

**Step 1** Right-click the user.

The Properties dialog box appears.

**Step 2** Click the Dial-in tab, then click the Allow Access radio button (Figure C-9).
Configuring an External LDAP Server

Appendix C      Configuring an External Server for Authorization and Authentication

Configuring an External LDAP Server

Figure C-9      AD/LDAP User1 - Allow Access

Note
If you select the Control access through the Remote Access Policy option, then a value is not returned from the server, and the permissions that are enforced are based on the internal group policy settings of the ASASM.

Step 3
Create an attribute map to allow both an IPsec and AnyConnect connection, but deny a clientless SSL connection.

The following example shows how to create the map tunneling_protocols, and map the AD attribute msNPAllowDialin used by the Allow Access setting to the Cisco attribute Tunneling-Protocols using the map-name command, and add map values with the map-value command:

hostname(config)# ldap attribute-map tunneling_protocols
hostname(config-ldap-attribute-map)# map-name msNPAllowDialin Tunneling-Protocols
hostname(config-ldap-attribute-map)# map-value msNPAllowDialin FALSE 48
hostname(config-ldap-attribute-map)# map-value msNPAllowDialin TRUE 4

Step 4
Associate the LDAP attribute map to the AAA server.

The following example enters the aaa server host configuration mode for the host 10.1.1.2, in the AAA server group MS_LDAP, and associates the attribute map tunneling_protocols that you created in Step 2:

hostname(config)# aaa-server MS_LDAP host 10.1.1.2
hostname(config-aaa-server-host)# ldap-attribute-map tunneling_protocols

Step 5
Verify that the attribute map works as configured.

Step 6
Try connections using clientless SSL, the AnyConnect client, and the IPsec client. The clientless and AnyConnect connections should fail, and the user should be informed that an unauthorized connection mechanism was the reason for the failed connection. The IPsec client should connect because IPsec is an allowed tunneling protocol according to the attribute map (see Figure C-10 and Figure C-11).
Enforcing Logon Hours and Time-of-Day Rules

The following example shows how to configure and enforce the hours that a clientless SSL user (such as a business partner) is allowed to access the network.

On the AD server, use the Office field to enter the name of the partner, which uses the physicalDeliveryOfficeName attribute. Then we create an attribute map on the ASASM to map that attribute to the Cisco attribute Access-Hours. During authentication, the ASASM retrieves the value of physicalDeliveryOfficeName and maps it to Access-Hours.

To configure the user attributes on the AD/LDAP server, perform the following steps:

**Step 1** Select the user, and right-click **Properties**.

The Properties dialog box appears (see Figure C-12).

**Step 2** Click the **General** tab.
Step 3  Create an attribute map.

The following example shows how to create the attribute map access_hours and map the AD attribute physicalDeliveryOfficeName used by the Office field to the Cisco attribute Access-Hours.

```
hostname(config)# ldap attribute-map access_hours
hostname(config-ldap-attribute-map)# map-name physicalDeliveryOfficeName Access-Hours
```

Step 4  Associate the LDAP attribute map to the AAA server.

The following example enters the aaa server host configuration mode for the host 10.1.1.2, in the AAA server group MS_LDAP, and associates the attribute map access_hours that you created in Step 3:

```
hostname(config)# aaa-server MS_LDAP host 10.1.1.2
hostname(config-aaa-server-host)# ldap-attribute-map access_hours
```

Step 5  Configure time ranges for each value allowed on the server.

The following example configures Partner access hours from 9am to 5pm Monday through Friday:

```
hostname(config)# time-range Partner
hostname(config-time-range)# periodic weekdays 09:00 to 17:00
```
### Configuring an External RADIUS Server

This section presents an overview of the RADIUS configuration procedure and defines the Cisco RADIUS attributes. It includes the following topics:

- **Reviewing the RADIUS Configuration Procedure**, page C-27
- **ASASM RADIUS Authorization Attributes**, page C-27
- **ASASM IETF RADIUS Authorization Attributes**, page C-36
- **RADIUS Accounting Disconnect Reason Codes**, page C-37

#### Reviewing the RADIUS Configuration Procedure

This section describes the RADIUS configuration steps required to support authentication and authorization of ASASM users.

To set up the RADIUS server to interoperate with the ASASM, perform the following steps:

**Step 1** Load the ASASM attributes into the RADIUS server. The method you use to load the attributes depends on which type of RADIUS server you are using:

- If you are using Cisco ACS: the server already has these attributes integrated. You can skip this step.
- If you are using a FUNK RADIUS server: Cisco supplies a dictionary file that contains all the ASASM attributes. Obtain this dictionary file, cisco3k.dct, from the Cisco Download Software Center on Cisco.com or from the ASASM CD-ROM. Load the dictionary file on your server.
- For RADIUS servers from other vendors (for example, Microsoft Internet Authentication Service): you must manually define each ASASM attribute. To define an attribute, use the attribute name or number, type, value, and vendor code (3076). For a list of ASASM RADIUS authorization attributes and values, see Table C-7.

**Step 2** Set up the users or groups with the permissions and attributes to send during IPsec or SSL tunnel establishment.

#### ASASM RADIUS Authorization Attributes

Authorization refers to the process of enforcing permissions or attributes. A RADIUS server defined as an authentication server enforces permissions or attributes if they are configured. These attributes have vendor ID 3076.

Table C-7 lists the ASASM supported RADIUS attributes that can be used for user authorization.

---

**Note**

RADIUS attribute names do not contain the cVPN3000 prefix. Cisco Secure ACS 4.x supports this new nomenclature, but attribute names in pre-4.0 ACS releases still include the cVPN3000 prefix. The ASASMs enforce the RADIUS attributes based on attribute numeric ID, not attribute name. LDAP attributes are enforced by their name, not by the ID.

All attributes listed in Table C-7 are downstream attributes that are sent from the RADIUS server to the ASASM except for the following attribute numbers: 146, 150, 151, and 152. These attribute numbers are upstream attributes that are sent from the ASASM to the RADIUS server. RADIUS attributes 146 and
150 are sent from the ASA to the RADIUS server for authentication and authorization requests. All four previously listed attributes are sent from the ASA to the RADIUS server for accounting start, interim-update, and stop requests. Upstream RADIUS attributes 146, 150, 151, and 152 were introduced in ASA Version 8.4.3.

### Table C-7 ASASM Supported RADIUS Attributes and Values

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>VPN 3000</th>
<th>ASA</th>
<th>PIX</th>
<th>Attr. No.</th>
<th>Syntax/Type</th>
<th>Single or Multi-Valued</th>
<th>Description or Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access-Hours</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>1</td>
<td>String</td>
<td>Single</td>
<td>Name of the time range, for example, Business-hours</td>
</tr>
<tr>
<td>Simultaneous-Logins</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>2</td>
<td>Integer</td>
<td>Single</td>
<td>0 - 2147483647</td>
</tr>
<tr>
<td>Primary-DNS</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>5</td>
<td>String</td>
<td>Single</td>
<td>An IP address</td>
</tr>
<tr>
<td>Secondary-DNS</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>6</td>
<td>String</td>
<td>Single</td>
<td>An IP address</td>
</tr>
<tr>
<td>Primary-WINS</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>7</td>
<td>String</td>
<td>Single</td>
<td>An IP address</td>
</tr>
<tr>
<td>Secondary-WINS</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>8</td>
<td>String</td>
<td>Single</td>
<td>An IP address</td>
</tr>
<tr>
<td>SEP-Card-Assignment</td>
<td></td>
<td></td>
<td></td>
<td>9</td>
<td>Integer</td>
<td>Single</td>
<td>Not used</td>
</tr>
</tbody>
</table>
| Tunneling-Protocols             | Y        | Y   | Y   | 11        | Integer     | Single                 | 1 = PPTP  
2 = L2TP  
4 = IPSec (IKEv1)  
8 = L2TP/IPSec  
16 = WebVPN  
32 = SVC  
64 = IPSec (IKEv2)  
8 and 4 are mutually exclusive  
(0 - 11, 16 - 27, 32 - 43, 48 - 59 are legal values). |
| IPsec-Sec-Association           | Y        |     |     | 12        | String      | Single                 | Name of the security association |
| IPsec-Authentication            | Y        |     |     | 13        | Integer     | Single                 | 0 = None  
1 = RADIUS  
2 = LDAP (authorization only)  
3 = NT Domain  
4 = SDI  
5 = Internal  
6 = RADIUS with Expiry  
7 = Kerberos/Active Directory |
| Banner1                         | Y        | Y   | Y   | 15        | String      | Single                 | Banner string to display for Cisco VPN remote access sessions: IPSec IKEv1, AnyConnect SSL-TLS/DTLS/IKEv2, and Clientless SSL |
| IPsec-Allow-Passwd-Store        | Y        | Y   | Y   | 16        | Boolean     | Single                 | 0 = Disabled  
1 = Enabled |
### Table C-7  ASASM Supported RADIUS Attributes and Values (continued)

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>VPN 3000</th>
<th>ASA</th>
<th>PIX</th>
<th>Attr. No.</th>
<th>Syntax/Type</th>
<th>Single or Multi-Valued</th>
<th>Description or Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use-Client-Address</td>
<td>Y</td>
<td></td>
<td></td>
<td>17</td>
<td>Boolean</td>
<td>Single</td>
<td>0 = Disabled, 1 = Enabled</td>
</tr>
<tr>
<td>PPTP-Encryption</td>
<td>Y</td>
<td></td>
<td></td>
<td>20</td>
<td>Integer</td>
<td>Single</td>
<td>Bitmap: 1 = Encryption required, 2 = 40 bits, 4 = 128 bits, 8 = Stateless-Required, 15 = 40/128-Encr/Stateless-Req</td>
</tr>
<tr>
<td>L2TP-Encryption</td>
<td>Y</td>
<td></td>
<td></td>
<td>21</td>
<td>Integer</td>
<td>Single</td>
<td>Bitmap: 1 = Encryption required, 2 = 40 bits, 4 = 128 bits, 8 = Stateless-Req, 15 = 40/128-Encr/Stateless-Req</td>
</tr>
</tbody>
</table>
| Group-Policy                   | Y Y      |     |     | 25        | String      | Single                 | Sets the group policy for the remote access VPN session. For versions 8.2 and later, use this attribute instead of IETF-Radius-Class. You can use one of the three following formats:  
- *group policy name*  
- *OU=*group policy name*  
- *OU=*group policy name; |
| IPsec-Split-Tunnel-List        | Y Y Y    |     |     | 27        | String      | Single                 | Specifies the name of the network/access list that describes the split tunnel inclusion list. |
| IPsec-Default-Domain           | Y Y Y    |     |     | 28        | String      | Single                 | Specifies the single default domain name to send to the client (1-255 characters). |
| IPsec-Split-DNS-Names          | Y Y Y    |     |     | 29        | String      | Single                 | Specifies the list of secondary domain names to send to the client (1-255 characters). |
| IPsec-Tunnel-Type              | Y Y Y    |     |     | 30        | Integer     | Single                 | 1 = LAN-to-LAN, 2 = Remote access |
| IPsec-Mode-Config              | Y Y Y    |     |     | 31        | Boolean     | Single                 | 0 = Disabled, 1 = Enabled |
| IPsec-User-Group-Lock          | Y        |     |     | 33        | Boolean     | Single                 | 0 = Disabled, 1 = Enabled |
**Table C-7 ASASM Supported RADIUS Attributes and Values (continued)**

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>VPN 3000</th>
<th>ASA</th>
<th>PIX</th>
<th>Attr. No.</th>
<th>Syntax/Type</th>
<th>Single or Multi-Valued</th>
<th>Description or Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPsec-Over-UDP</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>34</td>
<td>Boolean</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>IPsec-Over-UDP-Port</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>35</td>
<td>Integer</td>
<td>Single</td>
<td>4001 - 49151. The default is10000.</td>
</tr>
<tr>
<td>Banner2</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>36</td>
<td>String</td>
<td>Single</td>
<td>Banner string to display for Cisco VPN remote access sessions: IPsec IKEv1, AnyConnect SSL-TLS/DTLS/IKEv2, and Clientless SSL. The Banner2 string is concatenated to the Banner1 string, if configured.</td>
</tr>
<tr>
<td>PPTP-MPPC-Compression</td>
<td>Y</td>
<td></td>
<td></td>
<td>37</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>L2TP-MPPC-Compression</td>
<td>Y</td>
<td></td>
<td></td>
<td>38</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>IPsec-IP-Compression</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>39</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>IPsec-IKE-Peer-ID-Check</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>40</td>
<td>Integer</td>
<td>Single</td>
<td>1 = Required 2 = If supported by peer certificate 3 = Do not check</td>
</tr>
<tr>
<td>IKE-Keep-Alives</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>41</td>
<td>Boolean</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>IPsec-Auth-On-Rekey</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>42</td>
<td>Boolean</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>Required-Client-Firewall-Vendor-Code</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>45</td>
<td>Integer</td>
<td>Single</td>
<td>1 = Cisco Systems (with Cisco Integrated Client) 2 = Zone Labs 3 = NetworkICE 4 = Sygate 5 = Cisco Systems (with Cisco Intrusion Prevention Security Agent)</td>
</tr>
</tbody>
</table>
### Table C-7  ASASM Supported RADIUS Attributes and Values (continued)

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>VPN 3000</th>
<th>ASA</th>
<th>PIX</th>
<th>Attr. No.</th>
<th>Syntax/Type</th>
<th>Single or Multi-Valued</th>
<th>Description or Value</th>
</tr>
</thead>
</table>
| Required-Client-Firewall-Product-Code               | Y         | Y   | Y   | 46        | Integer     | Single                 | Cisco Systems Products:  
|                                                     |           |     |     |           |             |                        | 1 = Cisco Intrusion Prevention Security Agent or Cisco Integrated Client (CIC)  
|                                                     |           |     |     |           |             |                        | Zone Labs Products:  
|                                                     |           |     |     |           |             |                        | 1 = Zone Alarm  
|                                                     |           |     |     |           |             |                        | 2 = Zone AlarmPro  
|                                                     |           |     |     |           |             |                        | 3 = Zone Labs Integrity  
|                                                     |           |     |     |           |             |                        | NetworkICE Product:  
|                                                     |           |     |     |           |             |                        | 1 = BlackIce Defender/Agent  
|                                                     |           |     |     |           |             |                        | Sygate Products:  
|                                                     |           |     |     |           |             |                        | 1 = Personal Firewall  
|                                                     |           |     |     |           |             |                        | 2 = Personal Firewall Pro  
|                                                     |           |     |     |           |             |                        | 3 = Security Agent |
| Required-Client-Firewall-Description                | Y         | Y   | Y   | 47        | String      | Single                 | String                                                                 |
| Require-HW-Client-Auth                              | Y         | Y   | Y   | 48        | Boolean     | Single                 | 0 = Disabled  
|                                                     |           |     |     |           |             |                        | 1 = Enabled  |
| Required-Individual-User-Auth                       | Y         | Y   | Y   | 49        | Integer     | Single                 | 0 = Disabled  
|                                                     |           |     |     |           |             |                        | 1 = Enabled  |
| Authenticated-User-Idle-Timeout                      | Y         | Y   | Y   | 50        | Integer     | Single                 | 1-35791394 minutes                                                                 |
| Cisco-IP-Phone-Bypass                               | Y         | Y   | Y   | 51        | Integer     | Single                 | 0 = No split tunneling  
|                                                     |           |     |     |           |             |                        | 1 = Split tunneling  
|                                                     |           |     |     |           |             |                        | 2 = Local LAN permitted  |
| IPsec-Split-Tunneling-Policy                        | Y         | Y   | Y   | 55        | Integer     | Single                 | 0 = None  
|                                                     |           |     |     |           |             |                        | 1 = Policy defined by remote FW Are-You-There (AYT)  
|                                                     |           |     |     |           |             |                        | 2 = Policy pushed CPP  
|                                                     |           |     |     |           |             |                        | 4 = Policy from server  |
| IPsec-Required-Client-Firewall-Capability           | Y         | Y   | Y   | 56        | Integer     | Single                 | 0 = None  
|                                                     |           |     |     |           |             |                        | 1 = Policy defined by remote FW Are-You-There (AYT)  
|                                                     |           |     |     |           |             |                        | 2 = Policy pushed CPP  
|                                                     |           |     |     |           |             |                        | 4 = Policy from server  |
| IPsec-Client-Firewall-Filter-Name                   | Y         |     |     | 57        | String      | Single                 | Specifies the name of the filter to be pushed to the client as firewall policy |
| IPsec-Client-Firewall-Filter-Optional               | Y         | Y   | Y   | 58        | Integer     | Single                 | 0 = Required  
|                                                     |           |     |     |           |             |                        | 1 = Optional  |
| IPsec-Backup-Servers                                | Y         | Y   | Y   | 59        | String      | Single                 | 1 = Use Client-Configured list  
|                                                     |           |     |     |           |             |                        | 2 = Disable and clear client list  
|                                                     |           |     |     |           |             |                        | 3 = Use Backup Server list  |
| IPsec-Backup-Server-List                           | Y         | Y   | Y   | 60        | String      | Single                 | Server Addresses (space delimited)  |
### Table C-7  ASASM Supported RADIUS Attributes and Values (continued)

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>VPN 3000</th>
<th>ASA</th>
<th>PIX</th>
<th>Attr. No.</th>
<th>Syntax/Type</th>
<th>Single or Multi-Valued</th>
<th>Description or Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>DHCP-Network-Scope</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>61</td>
<td>String</td>
<td>Single</td>
<td>IP Address</td>
</tr>
<tr>
<td>Intercept-DHCP-Configure-Msg</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>62</td>
<td>Boolean</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>MS-Client-Subnet-Mask</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>63</td>
<td>Boolean</td>
<td>Single</td>
<td>An IP address</td>
</tr>
<tr>
<td>Allow-Network-Extension-Mode</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>64</td>
<td>Boolean</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>Authorization-Type</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>65</td>
<td>Integer</td>
<td>Single</td>
<td>0 = None 1 = RADIUS 2 = LDAP</td>
</tr>
<tr>
<td>Authorization-Required</td>
<td>Y</td>
<td></td>
<td></td>
<td>66</td>
<td>Integer</td>
<td>Single</td>
<td>0 = No 1 = Yes</td>
</tr>
<tr>
<td>Authorization-DN-Field</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>67</td>
<td>String</td>
<td>Single</td>
<td>Possible values: UID, OU, O, CN, L, SP, C, EA, T, N, GN, SN, I, GENQ, DNQ, SER, use-entire-name</td>
</tr>
<tr>
<td>IKE-KeepAlive-Confidence-Interval</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>68</td>
<td>Integer</td>
<td>Single</td>
<td>10 - 300 seconds</td>
</tr>
<tr>
<td>WebVPN-Content-Filter-Parameters</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>69</td>
<td>Integer</td>
<td>Single</td>
<td>1 = Java ActiveX 2 = Java Script 4 = Image 8 = Cookies in images</td>
</tr>
<tr>
<td>WebVPN-URL-List</td>
<td>Y</td>
<td></td>
<td></td>
<td>71</td>
<td>String</td>
<td>Single</td>
<td>URL-List name</td>
</tr>
<tr>
<td>WebVPN-Port-Forward-List</td>
<td>Y</td>
<td></td>
<td></td>
<td>72</td>
<td>String</td>
<td>Single</td>
<td>Port-Forward list name</td>
</tr>
<tr>
<td>WebVPN-Access-List</td>
<td>Y</td>
<td></td>
<td></td>
<td>73</td>
<td>String</td>
<td>Single</td>
<td>Access-List name</td>
</tr>
<tr>
<td>Cisco-LEAP-Bypass</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>75</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>WebVPN-Homepage</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>76</td>
<td>String</td>
<td>Single</td>
<td>A URL such as <a href="http://example-example.com">http://example-example.com</a></td>
</tr>
<tr>
<td>Client-Type-Version-Limiting</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>77</td>
<td>String</td>
<td>Single</td>
<td>IPsec VPN version number string</td>
</tr>
<tr>
<td>WebVPN-Port-Forwarding-Name</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>79</td>
<td>String</td>
<td>Single</td>
<td>String name (example, “Corporate-Apps”). This text replaces the default string, “Application Access,” on the clientless portal home page.</td>
</tr>
<tr>
<td>IE-Proxy-Server</td>
<td>Y</td>
<td></td>
<td></td>
<td>80</td>
<td>String</td>
<td>Single</td>
<td>IP address</td>
</tr>
<tr>
<td>IE-Proxy-Server-Policy</td>
<td>Y</td>
<td></td>
<td></td>
<td>81</td>
<td>Integer</td>
<td>Single</td>
<td>1 = No Modify 2 = No Proxy 3 = Auto detect 4 = Use Concentrator Setting</td>
</tr>
</tbody>
</table>
### Table C-7  ASASM Supported RADIUS Attributes and Values (continued)

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>VPN 3000</th>
<th>ASA</th>
<th>PIX</th>
<th>Attr. No.</th>
<th>Syntax/Type</th>
<th>Single or Multi-Valued</th>
<th>Description or Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>IE-Proxy-Exception-List</td>
<td>Y</td>
<td></td>
<td></td>
<td>82</td>
<td>String</td>
<td>Single</td>
<td>New line (\n) separated list of DNS domains</td>
</tr>
<tr>
<td>IE-Proxy-Bypass-Local</td>
<td>Y</td>
<td></td>
<td></td>
<td>83</td>
<td>Integer</td>
<td>Single</td>
<td>0 = None 1 = Local</td>
</tr>
<tr>
<td>IKE-Keepalive-Retry-Interval</td>
<td>Y  Y Y</td>
<td></td>
<td></td>
<td>84</td>
<td>Integer</td>
<td>Single</td>
<td>2 - 10 seconds</td>
</tr>
<tr>
<td>Tunnel-Group-Lock</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>85</td>
<td>String</td>
<td>Single</td>
<td>Name of the tunnel group or “none”</td>
</tr>
<tr>
<td>Access-List-Inbound</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>86</td>
<td>String</td>
<td>Single</td>
<td>Access list ID</td>
</tr>
<tr>
<td>Access-List-Outbound</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>87</td>
<td>String</td>
<td>Single</td>
<td>Access list ID</td>
</tr>
<tr>
<td>Perfect-Forward-Secrecy-Enable</td>
<td>Y  Y Y</td>
<td></td>
<td></td>
<td>88</td>
<td>Boolean</td>
<td>Single</td>
<td>0 = No 1 = Yes</td>
</tr>
<tr>
<td>NAC-Enable</td>
<td>Y</td>
<td></td>
<td></td>
<td>89</td>
<td>Integer</td>
<td>Single</td>
<td>0 = No 1 = Yes</td>
</tr>
<tr>
<td>NAC-Status-Query-Timer</td>
<td>Y</td>
<td></td>
<td></td>
<td>90</td>
<td>Integer</td>
<td>Single</td>
<td>30 - 1800 seconds</td>
</tr>
<tr>
<td>NAC-Revalidation-Timer</td>
<td>Y</td>
<td></td>
<td></td>
<td>91</td>
<td>Integer</td>
<td>Single</td>
<td>300 - 86400 seconds</td>
</tr>
<tr>
<td>NAC-Default-ACL</td>
<td>Y</td>
<td></td>
<td></td>
<td>92</td>
<td>String</td>
<td>Access list</td>
<td></td>
</tr>
<tr>
<td>WebVPN-URL-Entry-Enable</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>93</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>WebVPN-File-Access-Enable</td>
<td>Y  Y</td>
<td></td>
<td></td>
<td>94</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>WebVPN-File-Server-Entry-Enable</td>
<td>Y  Y</td>
<td></td>
<td></td>
<td>95</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>WebVPN-File-Server-Browsing-Enable</td>
<td>Y  Y</td>
<td></td>
<td></td>
<td>96</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>WebVPN-Port-Forwarding-Enable</td>
<td>Y  Y</td>
<td></td>
<td></td>
<td>97</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>WebVPN-Outlook-Exchange-Proxy-Enable</td>
<td>Y  Y</td>
<td></td>
<td></td>
<td>98</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>WebVPN-Port-Forwarding-HTTP-Proxy</td>
<td>Y  Y</td>
<td></td>
<td></td>
<td>99</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>WebVPN-Auto-Applet-Download-Enable</td>
<td>Y  Y</td>
<td></td>
<td></td>
<td>100</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>WebVPN-Citrix-Metaframe-Enable</td>
<td>Y  Y</td>
<td></td>
<td></td>
<td>101</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>WebVPN-Apply-ACL</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>102</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>WebVPN-SSL-VPN-Client-Enable</td>
<td>Y  Y</td>
<td></td>
<td></td>
<td>103</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
</tbody>
</table>
### Table C-7  ASASM Supported RADIUS Attributes and Values (continued)

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>VPN 3000</th>
<th>ASA</th>
<th>PIX</th>
<th>Attr. No.</th>
<th>Syntax/Type</th>
<th>Single or Multi-Valued</th>
<th>Description or Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>WebVPN-SSL-VPN-Client-Required</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>104</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>WebVPN-SSL-VPN-Client-Keep-Installation</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>105</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>SVC-Keepalive</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>107</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Off 15 - 600 seconds</td>
</tr>
<tr>
<td>SVC-DPD-Interval-Client</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>108</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Off 5 - 3600 seconds</td>
</tr>
<tr>
<td>SVC-DPD-Interval-Gateway</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>109</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Off 5 - 3600 seconds</td>
</tr>
<tr>
<td>SVC-Rekey-Time</td>
<td>Y</td>
<td></td>
<td></td>
<td>110</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 - 10080 minutes</td>
</tr>
<tr>
<td>WebVPN-Deny-Message</td>
<td>Y</td>
<td></td>
<td></td>
<td>116</td>
<td>String</td>
<td>Single</td>
<td>Valid string (up to 500 characters)</td>
</tr>
<tr>
<td>Extended-Authentication-On-Rekey</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>122</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>SVC-DTLS</td>
<td>Y</td>
<td></td>
<td></td>
<td>123</td>
<td>Integer</td>
<td>Single</td>
<td>0 = False 1 = True</td>
</tr>
<tr>
<td>SVC-MTU</td>
<td>Y</td>
<td></td>
<td></td>
<td>125</td>
<td>Integer</td>
<td>Single</td>
<td>MTU value 256 - 1406 in bytes</td>
</tr>
<tr>
<td>SVC-Modules</td>
<td>Y</td>
<td></td>
<td></td>
<td>127</td>
<td>String</td>
<td>Single</td>
<td>String (name of a module)</td>
</tr>
<tr>
<td>SVC-Profiles</td>
<td>Y</td>
<td></td>
<td></td>
<td>128</td>
<td>String</td>
<td>Single</td>
<td>String (name of a profile)</td>
</tr>
<tr>
<td>SVC-Ask</td>
<td>Y</td>
<td></td>
<td></td>
<td>131</td>
<td>String</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled 3 = Enable default service 5 = Enable default clientless (2 and 4 not used)</td>
</tr>
<tr>
<td>SVC-Ask-Timeout</td>
<td>Y</td>
<td></td>
<td></td>
<td>132</td>
<td>Integer</td>
<td>Single</td>
<td>5 - 120 seconds</td>
</tr>
<tr>
<td>IE-Proxy-PAC-URL</td>
<td>Y</td>
<td></td>
<td></td>
<td>133</td>
<td>String</td>
<td>Single</td>
<td>PAC Address String</td>
</tr>
<tr>
<td>Strip-Realm</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>135</td>
<td>Boolean</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>Smart-Tunnel</td>
<td>Y</td>
<td></td>
<td></td>
<td>136</td>
<td>String</td>
<td>Single</td>
<td>Name of a Smart Tunnel</td>
</tr>
<tr>
<td>WebVPN-ActiveX-Relay</td>
<td>Y</td>
<td></td>
<td></td>
<td>137</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled Otherwise = Enabled</td>
</tr>
<tr>
<td>Smart-Tunnel-Auto</td>
<td>Y</td>
<td></td>
<td></td>
<td>138</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled 2 = AutoStart</td>
</tr>
</tbody>
</table>
### Table C-7  ASASM Supported RADIUS Attributes and Values (continued)

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>VPN 3000</th>
<th>ASA</th>
<th>PIX</th>
<th>Attr. No.</th>
<th>Syntax/Type</th>
<th>Single or Multi-Valued</th>
<th>Description or Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Smart-Tunnel-Auto-Signon-Enable</td>
<td>Y</td>
<td></td>
<td></td>
<td>139</td>
<td>String</td>
<td>Single</td>
<td>Name of a Smart Tunnel Auto Signon list appended by the domain name</td>
</tr>
<tr>
<td>VLAN</td>
<td>Y</td>
<td></td>
<td></td>
<td>140</td>
<td>Integer</td>
<td>Single</td>
<td>0 - 4094</td>
</tr>
<tr>
<td>NAC-Settings</td>
<td>Y</td>
<td></td>
<td></td>
<td>141</td>
<td>String</td>
<td>Single</td>
<td>Name of the NAC policy</td>
</tr>
<tr>
<td>Member-Of</td>
<td>Y Y</td>
<td></td>
<td></td>
<td>145</td>
<td>String</td>
<td>Single</td>
<td>Comma-delimited string, for example: Engineering, Sales An administrative attribute that can be used in dynamic access policies. It does not set a group policy.</td>
</tr>
<tr>
<td>Tunnel Group Name</td>
<td>Y Y</td>
<td></td>
<td></td>
<td>146</td>
<td>String</td>
<td>Single</td>
<td>1 - 253 characters</td>
</tr>
<tr>
<td>Client Type</td>
<td>Y Y</td>
<td></td>
<td></td>
<td>150</td>
<td>Integer</td>
<td>Single</td>
<td>1 = Cisco VPN Client (IKEv1) 2 = AnyConnect Client SSL VPN 3 = Clientless SSL VPN 4 = Cut-Through-Proxy 5 = L2TP/IPsec SSL VPN 6 = AnyConnect Client IPsec VPN (IKEv2)</td>
</tr>
<tr>
<td>Session Type</td>
<td>Y Y</td>
<td></td>
<td></td>
<td>151</td>
<td>Integer</td>
<td>Single</td>
<td>0 = None 1 = AnyConnect Client SSL VPN 2 = AnyConnect Client IPSec VPN (IKEv2) 3 = Clientless SSL VPN 4 = Clientless Email Proxy 5 = Cisco VPN Client (IKEv1) 6 = IKEv1 LAN-LAN 7 = IKEv2 LAN-LAN 8 = VPN Load Balancing</td>
</tr>
<tr>
<td>Session Subtype</td>
<td>Y Y</td>
<td></td>
<td></td>
<td>152</td>
<td>Integer</td>
<td>Single</td>
<td>0 = None 1 = Clientless 2 = Client 3 = Client Only Session Subtype applies only when the Session Type (151) attribute has the following values: 1, 2, 3, and 4.</td>
</tr>
<tr>
<td>Address-Pools</td>
<td>Y Y</td>
<td></td>
<td></td>
<td>217</td>
<td>String</td>
<td>Single</td>
<td>Name of IP local pool</td>
</tr>
<tr>
<td>IPv6-Address-Pools</td>
<td>Y</td>
<td></td>
<td></td>
<td>218</td>
<td>String</td>
<td>Single</td>
<td>Name of IP local pool-IPv6</td>
</tr>
</tbody>
</table>
### Table C-7  ASASM Supported RADIUS Attributes and Values (continued)

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>VPN 3000</th>
<th>ASA</th>
<th>PIX</th>
<th>Attr. No.</th>
<th>Syntax/Type</th>
<th>Single or Multi-Valued</th>
<th>Description or Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPv6-VPN-Filter</td>
<td>Y</td>
<td></td>
<td></td>
<td>219</td>
<td>String</td>
<td>Single</td>
<td>ACL value</td>
</tr>
<tr>
<td>Privilege-Level</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td>220</td>
<td>Integer</td>
<td>Single</td>
<td>An integer between 0 and 15.</td>
</tr>
<tr>
<td>WebVPN-Macro-Value1</td>
<td>Y</td>
<td></td>
<td></td>
<td>223</td>
<td>String</td>
<td>Single</td>
<td>Unbounded. For examples, see the SSL VPN Deployment Guide at the following URL:</td>
</tr>
<tr>
<td>WebVPN-Macro-Value2</td>
<td>Y</td>
<td></td>
<td></td>
<td>224</td>
<td>String</td>
<td>Single</td>
<td>Unbounded. For examples, see the SSL VPN Deployment Guide at the following URL:</td>
</tr>
</tbody>
</table>

### ASASM IETF RADIUS Authorization Attributes

Table C-8 lists the supported IETF RADIUS attributes.

### Table C-8  ASASM Supported IETF RADIUS Attributes and Values

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>VPN 3000</th>
<th>ASA</th>
<th>PIX</th>
<th>Attr. No.</th>
<th>Syntax/Type</th>
<th>Single or Multi-Valued</th>
<th>Description or Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>IETF-Radius-Class</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>25</td>
<td>Single</td>
<td>Single</td>
<td>For Versions 8.2.x and later, we recommend that you use the Group-Policy attribute (VSA 3076, #25) as described in Table C-7:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• group policy name</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• OU=group policy name</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• OU=group policy name</td>
</tr>
<tr>
<td>IETF-Radius-Filter-Id</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>11</td>
<td>String</td>
<td>Single</td>
<td>Access list name that is defined on the ASASM, which applies only to full tunnel IPsec and SSL VPN clients</td>
</tr>
<tr>
<td>IETF-Radius-Framed-IP-Address</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>n/a</td>
<td>String</td>
<td>Single</td>
<td>An IP address</td>
</tr>
<tr>
<td>IETF-Radius-Framed-IP-Netmask</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>n/a</td>
<td>String</td>
<td>Single</td>
<td>An IP address mask</td>
</tr>
<tr>
<td>IETF-Radius-Idle-Timeout</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>28</td>
<td>Integer</td>
<td>Single</td>
<td>Seconds</td>
</tr>
</tbody>
</table>
### Table C-8  ASASM Supported IETF RADIUS Attributes and Values

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Y</th>
<th>Y</th>
<th>Y</th>
<th>Integer</th>
<th>Single</th>
<th>Seconds</th>
<th>Possible Service Type Values:</th>
</tr>
</thead>
<tbody>
<tr>
<td>IETF-Radius-Service-Type</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>. Administrative — User is allowed access to configure prompt.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>. NAS-Prompt — User is allowed access to exec prompt.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>. remote-access — User is allowed network access.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Y</th>
<th>Y</th>
<th>Y</th>
<th>Integer</th>
<th>Single</th>
<th>Seconds</th>
</tr>
</thead>
<tbody>
<tr>
<td>IETF-Radius-Session-Timeout</td>
<td>27</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### RADIUS Accounting Disconnect Reason Codes

These codes are returned if the ASA encounters a disconnect when sending packets:

<table>
<thead>
<tr>
<th>Disconnect Reason Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACCT_DISC_USER_REQ = 1</td>
</tr>
<tr>
<td>ACCT_DISC_LOST_CARRIER = 2</td>
</tr>
<tr>
<td>ACCT_DISC_LOST_SERVICE = 3</td>
</tr>
<tr>
<td>ACCT_DISC_IDLE_TIMEOUT = 4</td>
</tr>
<tr>
<td>ACCT_DISC_SESS_TIMEOUT = 5</td>
</tr>
<tr>
<td>ACCT_DISC_ADMIN_RESET = 6</td>
</tr>
<tr>
<td>ACCT_DISC_ADMIN_REBOOT = 7</td>
</tr>
<tr>
<td>ACCT_DISC_PORT_ERROR = 8</td>
</tr>
<tr>
<td>ACCT_DISC_NAS_ERROR = 9</td>
</tr>
<tr>
<td>ACCT_DISC_NAS_REQUEST = 10</td>
</tr>
<tr>
<td>ACCT_DISC_NAS_REBOOT = 11</td>
</tr>
<tr>
<td>ACCT_DISC_PORT_UNNEEDED = 12</td>
</tr>
<tr>
<td>ACCT_DISC_PORT_PREEMPTED = 13</td>
</tr>
<tr>
<td>ACCT_DISC_PORT_SUSPENDED = 14</td>
</tr>
<tr>
<td>ACCT_DISC_SERV_UNAVAIL = 15</td>
</tr>
<tr>
<td>ACCT_DISC_CALLBACK = 16</td>
</tr>
<tr>
<td>ACCT_DISC_USER_ERROR = 17</td>
</tr>
<tr>
<td>ACCT_DISC_HOST_REQUEST = 18</td>
</tr>
<tr>
<td>ACCT_DISC_ADMIN_SHUTDOWN = 19</td>
</tr>
<tr>
<td>ACCT_DISC_SA_EXPIRED = 21</td>
</tr>
<tr>
<td>ACCT_DISC_MAX_REASONS = 22</td>
</tr>
</tbody>
</table>
Configuring an External TACACS+ Server

The ASASM provides support for TACACS+ attributes. TACACS+ separates the functions of authentication, authorization, and accounting. The protocol supports two types of attributes: mandatory and optional. Both the server and client must understand a mandatory attribute, and the mandatory attribute must be applied to the user. An optional attribute may or may not be understood or used.

Note
To use TACACS+ attributes, make sure that you have enabled AAA services on the NAS.

Table C-9 lists supported TACACS+ authorization response attributes for cut-through-proxy connections. Table C-10 lists supported TACACS+ accounting attributes.

Table C-9  Supported TACACS+ Authorization Response Attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>acl</td>
<td>Identifies a locally configured access list to be applied to the connection.</td>
</tr>
<tr>
<td>idletime</td>
<td>Indicates the amount of inactivity in minutes that is allowed before the authenticated user session is terminated.</td>
</tr>
<tr>
<td>timeout</td>
<td>Specifies the absolute amount of time in minutes that authentication credentials remain active before the authenticated user session is terminated.</td>
</tr>
</tbody>
</table>

Table C-10  Supported TACACS+ Accounting Attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>bytes_in</td>
<td>Specifies the number of input bytes transferred during this connection (stop records only).</td>
</tr>
<tr>
<td>bytes_out</td>
<td>Specifies the number of output bytes transferred during this connection (stop records only).</td>
</tr>
<tr>
<td>cmd</td>
<td>Defines the command executed (command accounting only).</td>
</tr>
<tr>
<td>disc-cause</td>
<td>Indicates the numeric code that identifies the reason for disconnecting (stop records only).</td>
</tr>
<tr>
<td>elapsed_time</td>
<td>Defines the elapsed time in seconds for the connection (stop records only).</td>
</tr>
<tr>
<td>foreign_ip</td>
<td>Specifies the IP address of the client for tunnel connections. Defines the address on the lowest security interface for cut-through-proxy connections.</td>
</tr>
<tr>
<td>local_ip</td>
<td>Specifies the IP address that the client connected to for tunnel connections. Defines the address on the highest security interface for cut-through-proxy connections.</td>
</tr>
<tr>
<td>NAS port</td>
<td>Contains a session ID for the connection.</td>
</tr>
<tr>
<td>packs_in</td>
<td>Specifies the number of input packets transferred during this connection.</td>
</tr>
<tr>
<td>packs_out</td>
<td>Specifies the number of output packets transferred during this connection.</td>
</tr>
<tr>
<td>priv-level</td>
<td>Set to the user privilege level for command accounting requests or to 1 otherwise.</td>
</tr>
<tr>
<td>rem_iddr</td>
<td>Indicates the IP address of the client.</td>
</tr>
<tr>
<td>service</td>
<td>Specifies the service used. Always set to “shell” for command accounting only.</td>
</tr>
</tbody>
</table>
### Table C-10  Supported TACACS+ Accounting Attributes (continued)

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>task_id</td>
<td>Specifies a unique task ID for the accounting transaction.</td>
</tr>
<tr>
<td>username</td>
<td>Indicates the name of the user.</td>
</tr>
</tbody>
</table>
**3DES**
See DES.

### A

**AAA**
Authentication, authorization, and accounting. See also TACACS+ and RADIUS.

**ABR**
Area Border Router. In OSPF, a router with interfaces in multiple areas.

**ACE**
access control entry. Information entered into the configuration that lets you specify what type of traffic to permit or deny on an interface. By default, traffic that is not explicitly permitted is denied.

**Access Modes**
The ASASM CLI uses several command modes. The commands available in each mode vary. See also user EXEC mode, privileged EXEC mode, global configuration mode, command-specific configuration mode.

**ACL**
access control list. A collection of ACEs. An ACL lets you specify what type of traffic to allow on an interface. By default, traffic that is not explicitly permitted is denied. ACLs are usually applied to the interface which is the source of inbound traffic. See also rule, outbound ACL.

**ActiveX**
A set of object-oriented programming technologies and tools used to create mobile or portable programs. An ActiveX program is roughly equivalent to a Java applet.

**Address Resolution Protocol**
See ARP.

**address translation**
The translation of a network address and/or port to another network address/or port. See also IP address, interface PAT, NAT, PAT, Static PAT, xlate.

**AES**
Advanced Encryption Standard. A symmetric block cipher that can encrypt and decrypt information. The AES algorithm is capable of using cryptographic keys of 128, 192 and 256 bits to encrypt and decrypt data in blocks of 128 bits. See also DES.

**AH**
Authentication Header. An IP protocol (type 51) that can ensure data integrity, authentication, and replay detection. AH is embedded in the data to be protected (a full IP datagram, for example). AH can be used either by itself or with ESP. AH is an older IPsec protocol that is less important in most networks than ESP. AH provides authentication services but does not provide encryption services. It is provided to ensure compatibility with IPsec peers that do not support ESP, which provides both authentication and encryption. See also encryption and VPN. Refer to the RFC 2402.

**AIP**
Advanced Inspection and Prevention. For example, the AIP SSM or AIP SSC, which runs IPS software.
A record address

“A” stands for address, and refers to name-to-address mapped records in DNS.

APCF

Application Profile Customization Framework. Lets the security appliance handle nonstandard applications so that they render correctly over a clientless SSL VPN connection.

ARP

Address Resolution Protocol. A low-level TCP/IP protocol that maps a hardware address, or MAC address, to an IP address. An example hardware address is 00:00:a6:00:01:ba. The first three groups of characters (00:00:a6) identify the manufacturer; the rest of the characters (00:01:ba) identify the system card. ARP is defined in RFC 826.

ASA

Adaptive Security Algorithm. Used by the ASASM to perform inspections. ASA allows one-way (inside to outside) connections without an explicit configuration for each internal system and application. See also inspection engine.

ASA

adaptive ASASM.

ASDM

Adaptive Security Device Manager. An application for managing and configuring a single ASASM.

asymmetric encryption

Also called public key systems, asymmetric encryption allows anyone to obtain access to the public key of anyone else. Once the public key is accessed, you can send an encrypted message to that person using the public key. See also encryption, public key.

authentication

Cryptographic protocols and services that verify the identity of users and the integrity of data. One of the functions of the IPsec framework. Authentication establishes the integrity of the datastream and ensures that it is not tampered with in transit. It also provides confirmation about the origin of the datastream. See also AAA, encryption, and VPN.

Auto Applet Download

Automatically downloads the clientless SSL VPN port-forwarding applet when the user first logs in to clientless SSL VPN.

auto-signon

This command provides a single sign-on method for clientless SSL VPN users. It passes the clientless SSL VPN login credentials (username and password) to internal servers for authentication using NTLM authentication, basic authentication, or both.

B

backup server

IPsec backup servers let a VPN client connect to the central site when the primary security appliance is unavailable.

BGP

Border Gateway Protocol. BGP performs interdomain routing in TCP/IP networks. BGP is an Exterior Gateway Protocol, which means that it performs routing between multiple autonomous systems or domains and exchanges routing and access information with other BGP systems. The ASASM does not support BGP. See also EGP.

BLT stream

Bandwidth Limited Traffic stream. Stream or flow of packets whose bandwidth is constrained.

BOOTP

Bootstrap Protocol. Lets diskless workstations boot over the network as is described in RFC 951 and RFC 1542.

BPDU

Bridge Protocol Data Unit. Spanning-Tree Protocol hello packet that is sent out at configurable intervals to exchange information among bridges in the network. Protocol data unit is the OSI term for packet.
CA Certificate Authority, Certification Authority. A third-party entity that is responsible for issuing and revoking certificates. Each device with the public key of the CA can authenticate a device that has a certificate issued by the CA. The term CA also refers to software that provides CA services. See also certificate, CRL, public key, RA.

cache A temporary repository of information accumulated from previous task executions that can be reused, decreasing the time required to perform the tasks. Caching stores frequently reused objects in the system cache, which reduces the need to perform repeated rewriting and compressing of content.

CBC Cipher Block Chaining. A cryptographic technique that increases the encryption strength of an algorithm. CBC requires an initialization vector (IV) to start encryption. The IV is explicitly given in the IPsec packet.

certificate A signed cryptographic object that contains the identity of a user or device and the public key of the CA that issued the certificate. Certificates have an expiration date and may also be placed on a CRL if known to be compromised. Certificates also establish non-repudiation for IKE negotiation, which means that you can prove to a third party that IKE negotiation was completed with a specific peer.

CHAP Challenge Handshake Authentication Protocol.

CIFS Common Internet File System. It is a platform-independent file sharing system that provides users with network access to files, printers, and other machine resources. Microsoft implemented CIFS for networks of Windows computers, however, open source implementations of CIFS provide file access to servers running other operating systems, such as Linux, UNIX, and Mac OS X.

Citrix An application that virtualizes client-server applications and optimizes web applications.

CLI command-line interface. The primary interface for entering configuration and monitoring commands to the ASASM.

client/server computing Distributed computing (processing) network systems in which transaction responsibilities are divided into two parts: client (front end) and server (back end). Also called distributed computing. See also RPC.

Client update Lets you update revisions of clients to which the update applies; provide a URL or IP address from which to get the update; and, in the case of Windows clients, optionally notify users that they should update their VPN client version.

command-specific configuration mode From global configuration mode, some commands enter a command-specific configuration mode. All user EXEC, privileged EXEC, global configuration, and command-specific configuration commands are available in this mode. See also global configuration mode, privileged EXEC mode, user EXEC mode.

compression The process of encoding information using fewer bits or other information-bearing units than an unencoded representation would use. Compression can reduce the size of transferring packets and increase communication performance.

configuration, config, config file A file on the ASASM that represents the equivalent of settings, preferences, and properties administered by ASDM or the CLI.
<table>
<thead>
<tr>
<th><strong>Content Rewriting/Transformation</strong></th>
<th>Interprets and modifies applications so that they render correctly over a clientless SSL VPN connection.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>cookie</strong></td>
<td>A cookie is a object stored by a browser. Cookies contain information, such as user preferences, to persistent storage.</td>
</tr>
<tr>
<td><strong>CPU</strong></td>
<td>Central Processing Unit. Main processor.</td>
</tr>
<tr>
<td><strong>CRC</strong></td>
<td>Cyclical Redundancy Check. Error-checking technique in which the frame recipient calculates a remainder by dividing frame contents by a prime binary divisor and compares the calculated remainder to a value stored in the frame by the sending node.</td>
</tr>
<tr>
<td><strong>CRL</strong></td>
<td>Certificate Revocation List. A digitally signed message that lists all of the current but revoked certificates listed by a given CA. A CRL is analogous to a book of stolen charge card numbers that allow stores to reject bad credit cards. When certificates are revoked, they are added to a CRL. When you implement authentication using certificates, you can choose to use CRLs or not. Using CRLs lets you easily revoke certificates before they expire, but the CRL is generally only maintained by the CA or an RA. If you are using CRLs and the connection to the CA or RA is not available when authentication is requested, the authentication request will fail. See also <strong>CA</strong>, <strong>certificate</strong>, <strong>public key</strong>, <strong>RA</strong>.</td>
</tr>
<tr>
<td><strong>CRV</strong></td>
<td>Call Reference Value. Used by H.225.0 to distinguish call legs signaled between two entities.</td>
</tr>
<tr>
<td><strong>cryptography</strong></td>
<td>Encryption, authentication, integrity, keys and other services used for secure communication over networks. See also <strong>VPN</strong> and <strong>IPsec</strong>.</td>
</tr>
<tr>
<td><strong>crypto map</strong></td>
<td>A data structure with a unique name and sequence number that is used for configuring VPNs on the ASASM. A crypto map selects data flows that need security processing and defines the policy for these flows and the crypto peer that traffic needs to go to. A crypto map is applied to an interface. Crypto maps contain the ACLs, encryption standards, peers, and other parameters necessary to specify security policies for VPNs using <strong>IKE</strong> and <strong>IPsec</strong>. See also <strong>VPN</strong>.</td>
</tr>
<tr>
<td><strong>CTIQBE</strong></td>
<td>Computer Telephony Interface Quick Buffer Encoding. A protocol used in IP telephony between the Cisco CallManager and CTI TAPI and JTAPI applications. CTIQBE is used by the TAPI/JTAPI protocol inspection module and supports <strong>NAT</strong>, <strong>PAT</strong>, and bidirectional <strong>NAT</strong>. This protocol enables Cisco IP SoftPhone and other Cisco TAPI/JTAPI applications to communicate with Cisco CallManager for call setup and voice traffic across the ASASM.</td>
</tr>
<tr>
<td><strong>cut-through proxy</strong></td>
<td>Enables the ASASM to provide faster traffic flow after user authentication. The cut-through proxy challenges a user initially at the application layer. After the security appliance authenticates the user, it shifts the session flow and all traffic flows directly and quickly between the source and destination while maintaining session state information.</td>
</tr>
<tr>
<td><strong>D</strong></td>
<td><strong>data confidentiality</strong> Describes any method that manipulates data so that no attacker can read it. This is commonly achieved by data encryption and keys that are only available to the parties involved in the communication. <strong>data integrity</strong> Describes mechanisms that, through the use of encryption based on secret key or public key algorithms, allow the recipient of a piece of protected data to verify that the data has not been modified in transit.</td>
</tr>
</tbody>
</table>
data origin authentication

A security service where the receiver can verify that protected data could have originated only from the sender. This service requires a data integrity service plus a key distribution mechanism, where a secret key is shared only between the sender and receiver.

decryption

Application of a specific algorithm or cipher to encrypted data so as to render the data comprehensible to those who are authorized to see the information. See also encryption.

DES

Data encryption standard. DES was published in 1977 by the National Bureau of Standards and is a secret key encryption scheme based on the Lucifer algorithm from IBM. Cisco uses DES in classic crypto (40-bit and 56-bit key lengths), IPsec crypto (56-bit key), and 3DES (triple DES), which performs encryption three times using a 56-bit key. 3DES is more secure than DES but requires more processing for encryption and decryption. See also AES, ESP.

DHCP

Dynamic Host Configuration Protocol. Provides a mechanism for allocating IP addresses to hosts dynamically, so that addresses can be reused when hosts no longer need them and so that mobile computers, such as laptops, receive an IP address applicable to the LAN to which it is connected.

Diffie-Hellman

A public key cryptography protocol that allows two parties to establish a shared secret over insecure communications channels. Diffie-Hellman is used within IKE to establish session keys. Diffie-Hellman is a component of Oakley key exchange.

Diffie-Hellman Group 1, Group 2, Group 5, Group 7

Diffie-Hellman refers to a type of public key cryptography using asymmetric encryption based on large prime numbers to establish both Phase 1 and Phase 2 SAs. Group 1 provides a smaller prime number than Group 2 but may be the only version supported by some IPsec peers. Diffie-Hellman Group 5 uses a 1536-bit prime number, is the most secure, and is recommended for use with AES. Group 7 has an elliptical curve field size of 163 bits and is for use with the Movian VPN client, but works with any peer that supports Group 7 (ECC). See also VPN and encryption.

Note The group 7 command option was deprecated in ASA Version 8.0(4). Attempts to configure group 7 will generate an error message and use group 5 instead.

digital certificate

See certificate.

DMZ

See interface.

DN

Distinguished Name. Global, authoritative name of an entry in the OSI Directory (X.500).

DNS

Domain Name System (or Service). An Internet service that translates domain names into IP addresses.

DoS

Denial of Service. A type of network attack in which the goal is to render a network service unavailable.

DSL

digital subscriber line. Public network technology that delivers high bandwidth over conventional copper wiring at limited distances. DSL is provisioned via modem pairs, with one modem located at a central office and the other at the customer site. Because most DSL technologies do not use the whole bandwidth of the twisted pair, there is room remaining for a voice channel.

DSP

digital signal processor. A DSP segments a voice signal into frames and stores them in voice packets.

DSS

Digital Signature Standard. A digital signature algorithm designed by The US National Institute of Standards and Technology and based on public-key cryptography. DSS does not do user datagram encryption. DSS is a component in classic crypto, as well as the Redcreek IPsec card, but not in IPsec implemented in Cisco IOS software.
Dynamic NAT  See NAT and address translation.

Dynamic PAT  Dynamic Port Address Translation. Dynamic PAT lets multiple outbound sessions appear to originate from a single IP address. With PAT enabled, the ASASM chooses a unique port number from the PAT IP address for each outbound translation slot (xlate). This feature is valuable when an ISP cannot allocate enough unique IP addresses for your outbound connections. The global pool addresses always come first, before a PAT address is used. See also NAT, Static PAT, and xlate.

ECHO  See ping, ICMP. See also inspection engine.

EGP  Exterior Gateway Protocol. Replaced by BGP. The ASASM does not support EGP. See also BGP.

EIGRP  Enhanced Interior Gateway Routing Protocol. The ASASM does not support EIGRP.

EMBLEM  Enterprise Management BaseLine Embedded Manageability. A syslog format designed to be consistent with the Cisco IOS system log format and is more compatible with CiscoWorks management applications.

encryption  Application of a specific algorithm or cipher to data so as to render the data incomprehensible to those unauthorized to see the information. See also decryption.

ESMTP  Extended SMTP. Extended version of SMTP that includes additional functionality, such as delivery notification and session delivery. ESMTP is described in RFC 1869, SMTP Service Extensions.

ESP  Encapsulating Security Payload. An IPsec protocol, ESP provides authentication and encryption services for establishing a secure tunnel over an insecure network. For more information, refer to RFCs 2406 and 1827.

fixup  See inspection engine.

Flash, Flash memory  A nonvolatile storage device used to store the configuration file when the ASASM is powered down.

FQDN/IP  Fully qualified domain name/IP address. IPsec parameter that identifies peers that are security gateways.
FragGuard  Provides IP fragment protection and performs full reassembly of all ICMP error messages and virtual reassembly of the remaining IP fragments that are routed through the ASASM.

FTP  File Transfer Protocol. Part of the TCP/IP protocol stack, used for transferring files between hosts.

G

GGSN  gateway GPRS support node. A wireless gateway that allows mobile cell phone users to access the public data network or specified private IP networks.

global configuration mode  Global configuration mode lets you change the ASASM configuration. All user EXEC, privileged EXEC, and global configuration commands are available in this mode. See also user EXEC mode, privileged EXEC mode, command-specific configuration mode.

GMT  Greenwich Mean Time. Replaced by UTC (Coordinated Universal Time) in 1967 as the world time standard.

GPRS  general packet radio service. A service defined and standardized by the European Telecommunication Standards Institute. GPRS is an IP-packet-based extension of GSM networks and provides mobile, wireless, data communications

GRE  Generic Routing Encapsulation described in RFCs 1701 and 1702. GRE is a tunneling protocol that can encapsulate a wide variety of protocol packet types inside IP tunnels, creating a virtual point-to-point link to routers at remote points over an IP network. By connecting multiprotocol subnetworks in a single-protocol backbone environment, IP tunneling using GRE allows network expansion across a single protocol backbone environment.

GSM  Global System for Mobile Communication. A digital, mobile, radio standard developed for mobile, wireless, voice communications.

GTP  GPRS tunneling protocol. GTP handles the flow of user packet data and signaling information between the SGSN and GGSN in a GPRS network. GTP is defined on both the Gn and Gp interfaces of a GPRS network.

H

H.225  A protocol used for TCP signaling in applications such as video conferencing. See also H.323 and inspection engine.

H.225.0  An ITU standard that governs H.225.0 session establishment and packetization. H.225.0 actually describes several different protocols: RAS, use of Q.931, and use of RTP.

H.245  An ITU standard that governs H.245 endpoint control.

H.320  Suite of ITU-T standard specifications for video conferencing over circuit-switched media, such as ISDN, fractional T-1, and switched-36 lines. Extensions of ITU-T standard H.320 enable video conferencing over LANs and other packet-switched networks, as well as video over the Internet.
H.323
Allows dissimilar communication devices to communicate with each other by using a standardized
communication protocol. H.323 defines a common set of CODECs, call setup and negotiating
procedures, and basic data transport methods.

H.323 RAS
Registration, admission, and status signaling protocol. Enables devices to perform registration,
admissions, bandwidth changes, and status and disengage procedures between VoIP gateway and the
gatekeeper.

H.450.2
Call transfer supplementary service for H.323.

H.450.3
Call diversion supplementary service for H.323.

Hash, Hash
Algorithm
A hash algorithm is a one-way function that operates on a message of arbitrary length to create a
fixed-length message digest used by cryptographic services to ensure its data integrity. MD5 has a
smaller digest and is considered to be slightly faster than SHA-1. Cisco uses both SHA-1 and MD5
hashes within our implementation of the IPsec framework. See also encryption, HMAC, and VPN.

headend
A firewall, concentrator, or other host that serves as the entry point into a private network for VPN
client connections over the public network. See also ISP and VPN.

HMAC
A mechanism for message authentication using cryptographic hashes such as SHA-1 and MD5.

host
The name for any device on a TCP/IP network that has an IP address. See also network and node.

host/network
An IP address and netmask used with other information to identify a single host or network subnet for
ASASM configuration, such as an address translation (xlate) or ACE.

HTTP
Hypertext Transfer Protocol. A protocol used by browsers and web servers to transfer files. When a
user views a web page, the browser can use HTTP to request and receive the files used by the web
page. HTTP transmissions are not encrypted.

HTTPS
Hypertext Transfer Protocol Secure. An SSL-encrypted version of HTTP.

I
IANA
Internet Assigned Number Authority. Assigns all port and protocol numbers for use on the Internet.

ICMP
Internet Control Message Protocol. Network-layer Internet protocol that reports errors and provides
other information relevant to IP packet processing.

IDS
Intrusion Detection System. A method of detecting malicious network activity by signatures and then
implementing a policy for that signature.

IETF
The Internet Engineering Task Force. A technical standards organization that develops RFC
documents defining protocols for the Internet.

IGMP
Internet Group Management Protocol. IGMP is a protocol used by IPv4 systems to report IP multicast
memberships to neighboring multicast routers.
IKE
Internet Key Exchange. IKE establishes a shared security policy and authenticates keys for services (such as IPsec) that require keys. Before any IPsec traffic can be passed, each ASASM must verify the identity of its peer. Identification can be done by manually entering preshared keys into both hosts or by a CA service. IKE is a hybrid protocol that uses part Oakley and part of another protocol suite called SKEME inside the ISAKMP framework. IKE (formerly known as ISAKMP/Oakley) is defined in RFC 2409.

IKE Extended Authentication
IKE Extended Authenticate (Xauth) is implemented per the IETF draft-ietf-ipsec-isakmp-xauth-04.txt (extended authentication). This protocol provides the capability of authenticating a user within IKE using TACACS+ or RADIUS.

IKE Mode Configuration
IKE Mode Configuration is implemented per the IETF draft-ietf-ipsec-isakmp-mode-cfg-04.txt. IKE Mode Configuration provides a method for a security gateway to download an IP address (and other network level configuration) to the VPN client as part of an IKE negotiation.

ILS
Internet Locator Service. ILS is based on LDAP and is ILSv2 compliant. ILS was developed by Microsoft for use with its NetMeeting, SiteServer, and Active Directory products.

IMAP
Internet Message Access Protocol. Method of accessing e-mail or bulletin board messages kept on a mail server that can be shared. IMAP permits client e-mail applications to access remote message stores as if they were local without actually transferring the message.

implicit rule
An access rule automatically created by the ASASM based on default rules or as a result of user-defined rules.

IMSI
International Mobile Subscriber Identity. One of two components of a GTP tunnel ID, the other being the NSAPI. See also NSAPI.

inside
The first interface, usually port 1, that connects your internal, trusted network protected by the ASASM. See also interface, interface name.

inspection engine
The ASASM inspects certain application-level protocols to identify the location of embedded addressing information in traffic. Inspection allows NAT to translate these embedded addresses and to update any checksum or other fields that are affected by the translation. Because many protocols open secondary TCP or UDP ports, each application inspection engine also monitors sessions to determine the port numbers for secondary channels. The initial session on a well-known port is used to negotiate dynamically assigned port numbers. The application inspection engine monitors these sessions, identifies the dynamic port assignments, and permits data exchange on these ports for the duration of the specific session. Some of the protocols that the ASASM can inspect are CTIQBE, FTP, H.323, HTTP, MGCP, SMTP, and SNMP.

interface
The physical connection between a particular network and a ASASM.

interface IP address
The IP address of the ASASM network interface. Each interface IP address must be unique. Two or more interfaces must not be given the same IP address or IP addresses that are on the same IP network.

interface name
Human-readable name assigned to the ASASM network interface. The inside interface default name is “inside” and the outside interface default name is “outside.” See also inside and outside.

interface PAT
The use of PAT where the PAT IP address is also the IP address of the outside interface. See Dynamic PAT, Static PAT.

Internet
The global network that uses IP. Not a LAN. See also intranet.
intranet  Intranetwork. A LAN that uses IP. See also network and Internet.

IP  Internet Protocol. IP protocols are the most popular nonproprietary protocols because they can be used to communicate across any set of interconnected networks and are equally well suited for LAN and WAN communications.

IPS  Intrusion Prevention Service. An in-line, deep-packet inspection-based solution that helps mitigate a wide range of network attacks.

IP address  An IP protocol address. A ASASM interface ip_address. IP version 4 addresses are 32 bits in length. This address space is used to designate the network number, optional subnetwork number, and a host number. The 32 bits are grouped into four octets (8 binary bits), represented by 4 decimal numbers separated by periods, or dots. The meaning of each of the four octets is determined by their use in a particular network.

IP pool  A range of local IP addresses specified by a name, and a range with a starting IP address and an ending address. IP pools are used by DHCP and VPNs to assign local IP addresses to clients on the inside interface.

IPsec  IP Security. A framework of open standards that provides data confidentiality, data integrity, and data authentication between participating peers. IPsec provides these security services at the IP layer. IPsec uses IKE to handle the negotiation of protocols and algorithms based on local policy and to generate the encryption and authentication keys to be used by IPsec. IPsec can protect one or more data flows between a pair of hosts, between a pair of security gateways, or between a security gateway and a host.

IPsec Phase 1  The first phase of negotiating IPsec, includes the key exchange and the ISAKMP portions of IPsec.

IPsec Phase 2  The second phase of negotiating IPsec. Phase 2 determines the type of encryption rules used for payload, the source and destination that will be used for encryption, the definition of interesting traffic according to access lists, and the IPsec peer. IPsec is applied to the interface in Phase 2.

IPsec transform set  A transform set specifies the IPsec protocol, encryption algorithm, and hash algorithm to use on traffic matching the IPsec policy. A transform describes a security protocol (AH or ESP) with its corresponding algorithms. The IPsec protocol used in almost all transform sets is ESP with the DES algorithm and HMAC-SHA for authentication.

ISAKMP  Internet Security Association and Key Management Protocol. A protocol framework that defines payload formats, the mechanics of implementing a key exchange protocol, and the negotiation of a security association. See IKE.

ISP  Internet Service Provider. An organization that provides connection to the Internet via their services, such as modem dial in over telephone voice lines or DSL.

J  

JTAPI  Java Telephony Application Programming Interface. A Java-based API supporting telephony functions. See also TAPI.
K

key
A data object used for encryption, decryption, or authentication.

L

L2TP
Layer Two Tunneling Protocol. An IETF standards track protocol defined in RFC 2661 that provides
channeling of PPP. L2TP is an extension to the PPP. L2TP merges the older Cisco Layer Two
Forwarding (L2F) protocol with PPTP. L2TP can be used with IPsec encryption and is considered
more secure against attack than PPTP.

LAN
Local area network. A network residing in one location, such as a single building or campus. See also
Internet, intranet, and network.

layer, layers
Networking models implement layers with which different protocols are associated. The most
common networking model is the OSI model, which consists of the following seven layers, in order:
physical, data link, network, transport, session, presentation, and application.

LCN
Logical channel number.

LDAP
Lightweight Directory Access Protocol. LDAP provides management and browser applications with
access to X.500 directories.

M

mask
A 32-bit mask that shows how an Internet address is divided into network, subnet, and host parts. The
mask has ones in the bit positions to be used for the network and subnet parts, and zeros for the host
part. The mask should contain at least the standard network portion, and the subnet field should be
contiguous with the network portion.

MCR
See multicast.

MC router
Multicast (MC) routers route multicast data transmissions to the hosts on each LAN in an internetwork
that are registered to receive specific multimedia or other broadcasts. See also multicast.

MD5
Message Digest 5. A one-way hashing algorithm that produces a 128-bit hash. Both MD5 and SHA-1
are variations on MD4 and are designed to strengthen the security of the MD4 hashing algorithm.
SHA-1 is more secure than MD4 and MD5. Cisco uses hashes for authentication within the IPsec
framework. Also used for message authentication in SNMP v.2. MD5 verifies the integrity of the
communication, authenticates the origin, and checks for timeliness. MD5 has a smaller digest and is
considered to be slightly faster than SHA-1.

MDI
media dependent interface.

MDIX
media dependent interface crossover.

message digest
A message digest is created by a hash algorithm, such as MD5 or SHA-1, that is used for ensuring
message integrity.
MGCP
Media Gateway Control Protocol. Media Gateway Control Protocol is a protocol for the control of VoIP calls by external call-control elements known as media gateway controllers or call agents. MGCP merges the IPDC and SGCP protocols.

Mode
See Access Modes.

Mode Config
See IKE Mode Configuration.

Modular Policy Framework
A means of configuring ASASM features in a manner similar to Cisco IOS software Modular QoS CLI.

MS
mobile station. Refers generically to any mobile device, such as a mobile handset or computer, that is used to access network services. GPRS networks support three classes of MS, which describe the type of operation supported within the GPRS and the GSM mobile wireless networks. For example, a Class A MS supports simultaneous operation of GPRS and GSM services.

MS-CHAP
Microsoft CHAP.

MTU
maximum transmission unit. The maximum number of bytes in a packet that can flow efficiently across the network with best response time. For Ethernet, the default MTU is 1500 bytes, but each network can have different values, with serial connections having the smallest values. The MTU is described in RFC 1191.

multicast
Refers to a network addressing method in which the source transmits a packet to multiple destinations, a multicast group, simultaneously. See also PIM, SMR.

N

N2H2
A third-party, policy-oriented filtering application that works with the ASASM to control user web access. N2H2 can filter HTTP requests based on the destination hostname, destination IP address, username, and password. The N2H2 corporation was acquired by Secure Computing in October, 2003.

NAT
Network Address Translation. Mechanism for reducing the need for globally unique IP addresses. NAT allows an organization with addresses that are not globally unique to connect to the Internet by translating those addresses into a globally routable address space.

NEM
Network Extension Mode. Lets VPN hardware clients present a single, routable network to the remote private network over the VPN tunnel.

NetBIOS
Network Basic Input/Output System. A Microsoft protocol that supports Windows hostname registration, session management, and data transfer. The ASASM supports NetBIOS by performing NAT of the packets for NBNS UDP port 137 and NBDS UDP port 138.

netmask
See mask.

network
In the context of ASASM configuration, a network is a group of computing devices that share part of an IP address space and not a single host. A network consists of multiple nodes or hosts. See also host, Internet, intranet, IP, LAN, and node.

NMS
network management system. System responsible for managing at least part of a network. An NMS is generally a reasonably powerful and well-equipped computer, such as an engineering workstation. NMSs communicate with agents to help keep track of network statistics and resources.
node  Devices such as routers and printers that would not normally be called hosts. See also host, network.

nonvolatile storage, memory  Storage or memory that, unlike RAM, retains its contents without power. Data in a nonvolatile storage device survives a power-off, power-on cycle.

NSAPI  network service access point identifier. One of two components of a GTP tunnel ID, the other component being the IMSI. See also IMSI.

NSSA  not-so-stubby-area. An OSPF feature described by RFC 1587. NSSA was first introduced in Cisco IOS software release 11.2. It is a nonproprietary extension of the existing stub area feature that allows the injection of external routes in a limited fashion into the stub area.


NTP  Network Time Protocol.

Oakley  A key exchange protocol that defines how to acquire authenticated keying material. The basic mechanism for Oakley is the Diffie-Hellman key exchange algorithm. Oakley is defined in RFC 2412.

object grouping  Simplifies access control by letting you apply access control statements to groups of network objects, such as protocol, services, hosts, and networks.

OSPF  Open Shortest Path First. OSPF is a routing protocol for IP networks. OSPF is a routing protocol widely deployed in large networks because of its efficient use of network bandwidth and its rapid convergence after changes in topology. The ASASM supports OSPF.

OU  Organizational Unit. An X.500 directory attribute.

outbound  Refers to traffic whose destination is on an interface with lower security than the source interface.

outbound ACL  An ACL applied to outbound traffic.

outside  The first interface, usually port 0, that connects to other untrusted networks outside the ASASM; the Internet. See also interface, interface name, outbound.

PAC  PPTP Access Concentrator. A device attached to one or more PSTN or ISDN lines capable of PPP operation and of handling the PPTP protocol. The PAC needs to implement TCP/IP to pass traffic to one or more PNSs. It may also tunnel non-IP protocols.

PAT  See Dynamic PAT, interface PAT, and Static PAT.

PDP  Packet Data Protocol.

Perfmon  The ASASM feature that gathers and reports a wide variety of feature statistics, such as connections/second, xlates/second, and so on.
**PFS**
Perfect Forwarding Secrecy. PFS enhances security by using a different security key for the IPsec Phase 1 and Phase 2 SAs. Without PFS, the same security key is used to establish SAs in both phases. PFS ensures that a given IPsec SA key was not derived from any other secret (like some other keys). In other words, if someone were to break a key, PFS ensures that the attacker would not be able to derive any other key. If PFS were not enabled, someone could hypothetically break the IKE SA secret key, copy all the IPsec protected data, and then use knowledge of the IKE SA secret to compromise the IPsec SA setup by this IKE SA. With PFS, breaking IKE would not give an attacker immediate access to IPsec. The attacker would have to break each IPsec SA individually.

**Phase 1**
See IPsec Phase 1.

**Phase 2**
See IPsec Phase 2.

**PIM**
Protocol Independent Multicast. PIM provides a scalable method for determining the best paths for distributing a specific multicast transmission to a group of hosts. Each host has registered using IGMP to receive the transmission. See also PIM-SM.

**PIM-SM**
Protocol Independent Multicast-Sparse Mode. With PIM-SM, which is the default for Cisco routers, when the source of a multicast transmission begins broadcasting, the traffic is forwarded from one MC router to the next, until the packets reach every registered host. See also PIM.

**ping**
An ICMP request sent by a host to determine if a second host is accessible.

**PIX**
Private Internet eXchange. The Cisco PIX 500 series ASASMs ranged from compact, plug-and-play desktop models for small/home offices to carrier-class gigabit models for the most demanding enterprise and service provider environments. Cisco PIX ASASMs provided robust, enterprise-class integrated network security services to create a strong multilayered defense for fast changing network environments. The PIX has been replaced by the Cisco ASA 5500 series.

**PKCS12**
A standard for the transfer of PKI-related data, such as private keys, certificates, and other data. Devices supporting this standard let administrators maintain a single set of personal identity information.

**PNS**
PPTP Network Server. A PNS is envisioned to operate on general-purpose computing/server platforms. The PNS handles the server side of PPTP. Because PPTP relies completely on TCP/IP and is independent of the interface hardware, the PNS may use any combination of IP interface hardware including LAN and WAN devices.

**Policy NAT**
Lets you identify local traffic for address translation by specifying the source and destination addresses (or ports) in an access list.

**POP**
Post Office Protocol. Protocol that client e-mail applications use to retrieve mail from a mail server.

**Pool**
See IP pool.

**Port**
A field in the packet headers of TCP and UDP protocols that identifies the higher level service which is the source or destination of the packet.

**PPP**
Point-to-Point Protocol. Developed for dial-up ISP access using analog phone lines and modems.

**PPPoE**
Point-to-Point Protocol over Ethernet. An IP protocol that encapsulates PPP packets and sends them over a local network or the internet to establish a connection to a host, usually between a client and an ISP.
PPTP  
Point-to-Point Tunneling Protocol. PPTP was introduced by Microsoft to provide secure remote access to Windows networks; however, because it is vulnerable to attack, PPTP is commonly used only when stronger security methods are not available or are not required. PPTP Ports are pptp, 1723/tcp, 1723/udp, and pptp. For more information about PPTP, see RFC 2637. See also PAC, PPTP GRE, PPTP GRE tunnel, PNS, PPTP session, and PPTP TCP.

PPTP GRE  
Version 1 of GRE for encapsulating PPP traffic.

PPTP GRE tunnel  
A tunnel defined by a PNS-PAC pair. The tunnel protocol is defined by a modified version of GRE. The tunnel carries PPP datagrams between the PAC and the PNS. Many sessions are multiplexed on a single tunnel. A control connection operating over TCP controls the establishment, release, and maintenance of sessions and of the tunnel itself.

PPTP session  
PPTP is connection-oriented. The PNS and PAC maintain the state for each user that is attached to a PAC. A session is created when an end-to-end PPP connection is attempted between a dial-up user and the PNS. The datagrams related to a session are sent over the tunnel between the PAC and PNS.

PPTP TCP  
Standard TCP session over which PPTP call control and management information is passed. The control session is logically associated with, but separate from, the sessions being tunneled through a PPTP tunnel.

preshared key  
A preshared key provides a method of IKE authentication that is suitable for networks with a limited, static number of IPsec peers. This method is limited in scalability because the key must be configured for each pair of IPsec peers. When a new IPsec peer is added to the network, the preshared key must be configured for every IPsec peer with which it communicates. Using certificates and CAs provides a more scalable method of IKE authentication.

primary, primary unit  
The ASASM normally operating when two units, a primary and secondary, are operating in failover mode.

privileged EXEC mode  
The highest privilege level at the ASA CLI. Any user EXEC mode command will work in privileged EXEC mode. The privileged EXEC mode prompt appears as follows after you enter the enable command:

```
hostname> enable
hostname#
```

See also command-specific configuration mode, global configuration mode, user EXEC mode.

protocol, protocol literals  
A standard that defines the exchange of packets between network nodes for communication. Protocols work together in layers. Protocols are specified in the ASASM configuration as part of defining a security policy by their literal values or port numbers. Possible ASASM protocol literal values are ahp, eigrp, esp, gre, icmp, igmp, igrp, ip, ipinip, ipsec, nos, ospf, pcp, snp, tcp, and udp.

Proxy-ARP  
Enables the ASASM to reply to an ARP request for IP addresses in the global pool. See also ARP.

public key  
A public key is one of a pair of keys that are generated by devices involved in public key infrastructure. Data encrypted with a public key can only be decrypted using the associated private key. When a private key is used to produce a digital signature, the receiver can use the public key of the sender to verify that the message was signed by the sender. These characteristics of key pairs provide a scalable and secure method of authentication over an insecure media, such as the Internet.
Q

QoS

quality of service. Measure of performance for a transmission system that reflects its transmission quality and service availability.

R

RA

Registration Authority. An authorized proxy for a CA. RAs can perform certificate enrollment and can issue CRLs. See also CA, certificate, public key.

RADIUS

Remote Authentication Dial-In User Service. RADIUS is a distributed client/server system that secures networks against unauthorized access. RFC 2058 and RFC 2059 define the RADIUS protocol standard. See also AAA and TACACS+.

refresh

Retrieve the running configuration from the ASASM and update the screen. The icon and the button perform the same function.

registration authority

See RA.

replay-detection

A security service where the receiver can reject old or duplicate packets to defeat replay attacks. Replay attacks rely on the attacker sending out older or duplicate packets to the receiver and the receiver thinking that the bogus traffic is legitimate. Replay-detection is done by using sequence numbers combined with authentication and is a standard feature of IPsec.

RFC

Request for Comments. RFC documents define protocols and standards for communications over the Internet. RFCs are developed and published by IETF.

RIP

Routing Information Protocol. Interior Gateway Protocol (IGP) supplied with UNIX BSD systems. The most common IGP in the Internet. RIP uses hop count as a routing metric.

RLLA

Reserved Link Local Address. Multicast addresses range from 224.0.0.0 to 239.255.255.255; however only the range 224.0.1.0 to 239.255.255.255 is available to users. The first part of the multicast address range, 224.0.0.0 to 224.0.0.255, is reserved and referred to as the RLLA. These addresses are unavailable.

route, routing

The path through a network.

routed firewall mode

In routed firewall mode, the ASASM is counted as a router hop in the network. It performs NAT between connected networks and can use OSPF or RIP. See also transparent firewall mode.

RPC

Remote Procedure Call. RPCs are procedure calls that are built or specified by clients and executed on servers, with the results returned over the network to the clients.

RSA

A public key cryptographic algorithm (named after its inventors, Rivest, Shamir, and Adelman) with a variable key length. The main weakness of RSA is that it is significantly slow to compute compared to popular secret-key algorithms, such as DES. The Cisco implementation of IKE uses a Diffie-Hellman exchange to get the secret keys. This exchange can be authenticated with RSA (or preshared keys). With the Diffie-Hellman exchange, the DES key never crosses the network (not even in encrypted form), which is not the case with the RSA encrypt and sign technique. RSA is not public domain, and must be licensed from RSA Data Security.
RSH  Remote Shell. A protocol that allows a user to execute commands on a remote system without having to log in to the system. For example, RSH can be used to remotely examine the status of a number of access servers without connecting to each communication server, executing the command, and then disconnecting from the communication server.

RTCP  RTP Control Protocol. Protocol that monitors the QoS of an IPv6 RTP connection and conveys information about the ongoing session. See also RTP.

RTP  Real-Time Transport Protocol. Commonly used with IP networks. RTP is designed to provide end-to-end network transport functions for applications transmitting real-time data, such as audio, video, or simulation data, over multicast or unicast network services. RTP provides such services as payload type identification, sequence numbering, timestamping, and delivery monitoring to real-time applications.

RTSP  Real Time Streaming Protocol. Enables the controlled delivery of real-time data, such as audio and video. RTSP is designed to work with established protocols, such as RTP and HTTP.

rule  Conditional statements added to the ASASM configuration to define security policy for a particular situation. See also ACE, ACL, NAT.

running configuration  The configuration currently running in RAM on the ASASM. The configuration that determines the operational characteristics of the ASASM.

S

SA  security association. An instance of security policy and keying material applied to a data flow. SAs are established in pairs by IPsec peers during both phases of IPsec. SAs specify the encryption algorithms and other security parameters used to create a secure tunnel. Phase 1 SAs (IKE SAs) establish a secure tunnel for negotiating Phase 2 SAs. Phase 2 SAs (IPsec SAs) establish the secure tunnel used for sending user data. Both IKE and IPsec use SAs, although SAs are independent of one another. IPsec SAs are unidirectional and they are unique in each security protocol. A set of SAs are needed for a protected data pipe, one per direction per protocol. For example, if you have a pipe that supports ESP between peers, one ESP SA is required for each direction. SAs are uniquely identified by destination (IPsec endpoint) address, security protocol (AH or ESP), and Security Parameter Index. IKE negotiates and establishes SAs on behalf of IPsec. A user can also establish IPsec SAs manually. An IKE SA is used by IKE only, and unlike the IPsec SA, it is bidirectional.

SCCP  Skinny Client Control Protocol. A Cisco-proprietary protocol used between Cisco Call Manager and Cisco VoIP phones.

SCEP  Simple Certificate Enrollment Protocol. A method of requesting and receiving (also known as enrolling) certificates from CAs.

SDP  Session Definition Protocol. An IETF protocol for the definition of Multimedia Services. SDP messages can be part of SGCP and MGCP messages.

secondary unit  The backup ASASM when two are operating in failover mode.

secret key  A secret key is a key shared only between the sender and receiver. See key, public key.
security context  You can partition a single ASASM into multiple virtual firewalls, known as security contexts. Each context is an independent firewall, with its own security policy, interfaces, and administrators. Multiple contexts are similar to having multiple stand-alone firewalls.

security services  See cryptography.

serial transmission  A method of data transmission in which the bits of a data character are transmitted sequentially over a single channel.

SGCP  Simple Gateway Control Protocol. Controls VoIP gateways by an external call control element (called a call-agent).

SGSN  Serving GPRS Support Node. The SGSN ensures mobility management, session management, and packet relaying functions.

SHA-1  Secure Hash Algorithm 1. SHA-1 [NIS94c] is a revision to SHA that was published in 1994. SHA is closely modeled after MD4 and produces a 160-bit digest. Because SHA produces a 160-bit digest, it is more resistant to brute-force attacks than 128-bit hashes (such as MD5), but it is slower. Secure Hash Algorithm 1 is a joint creation of the National Institute of Standards and Technology and the National Security Agency. This algorithm, like other hash algorithms, is used to generate a hash value, also known as a message digest, that acts like a CRC used in lower-layer protocols to ensure that message contents are not changed during transmission. SHA-1 is generally considered more secure than MD5.

SIP  Session Initiation Protocol. Enables call handling sessions, particularly two-party audio conferences, or calls. SIP works with SDP for call signaling. SDP specifies the ports for the media stream. Using SIP, the ASASM can support any SIP VoIP gateways and VoIP proxy servers.

site-to-site VPN  A site-to-site VPN is established between two IPsec peers that connect remote networks into a single VPN. In this type of VPN, neither IPsec peer is the destination nor source of user traffic. Instead, each IPsec peer provides encryption and authentication services for hosts on the LANs connected to each IPsec peer. The hosts on each LAN send and receive data through the secure tunnel established by the pair of IPsec peers.

SKEME  A key exchange protocol that defines how to derive authenticated keying material, with rapid key refreshment.

SMR  Stub Multicast Routing. SMR allows the ASASM to function as a stub router. A stub router is a device that acts as an IGMP proxy agent. IGMP is used to dynamically register specific hosts in a multicast group on a particular LAN with a multicast router. Multicast routers route multicast data transmissions to hosts that are registered to receive specific multimedia or other broadcasts. A stub router forwards IGMP messages between hosts and MC routers.

SMTP  Simple Mail Transfer Protocol. SMTP is an Internet protocol that supports email services.


split tunneling  Allows a remote VPN client simultaneous encrypted access to a private network and clear unencrypted access to the Internet. If you do not enable split tunneling, all traffic between the VPN client and the ASASM is sent through an IPsec tunnel. All traffic originating from the VPN client is sent to the outside interface through a tunnel, and client access to the Internet from its remote site is denied.
spoofing  A type of attack designed to foil network security mechanisms such as filters and access lists. A spoofing attack sends a packet that claims to be from an address from which it was not actually sent.

SQL*Net  Structured Query Language Protocol. An Oracle protocol used to communicate between client and server processes.

SSC  Security Services Card for the ASA 5505. For example, the AIP SSC.

SSH  Secure Shell. An application running on top of a reliable transport layer, such as TCP/IP, that provides strong authentication and encryption capabilities.

SSL  Secure Sockets Layer. A protocol that resides between the application layer and TCP/IP to provide transparent encryption of data traffic.

SSM  Security Services Module. For example, the AIP SSM or CSC SSM.

standby unit  See secondary unit.

stateful inspection  Network protocols maintain certain data, called state information, at each end of a network connection between two hosts. State information is necessary to implement the features of a protocol, such as guaranteed packet delivery, data sequencing, flow control, and transaction or session IDs. Some of the protocol state information is sent in each packet while each protocol is being used. For example, a browser connected to a web server uses HTTP and supporting TCP/IP protocols. Each protocol layer maintains state information in the packets it sends and receives. The ASASM and some other firewalls inspect the state information in each packet to verify that it is current and valid for every protocol it contains. This feature is called stateful inspection and is designed to create a powerful barrier to certain types of computer security threats.

Static PAT  Static Port Address Translation. Static PAT is a static address that also maps a local port to a global port. See also Dynamic PAT, NAT.

subnetmask  See mask.

T

TACACS+  Terminal Access Controller Access Control System Plus. A client-server protocol that supports AAA services, including command authorization. See also AAA, RADIUS.

TAPI  Telephony Application Programming Interface. A programming interface in Microsoft Windows that supports telephony functions.

TCP  Transmission Control Protocol. Connection-oriented transport layer protocol that provides reliable full-duplex data transmission.
TCP Intercept
With the TCP intercept feature, once the optional embryonic connection limit is reached, and until the embryonic connection count falls below this threshold, every SYN bound for the affected server is intercepted. For each SYN, the ASASM responds on behalf of the server with an empty SYN/ACK segment. The ASASM retains pertinent state information, drops the packet, and waits for the client acknowledgment. If the ACK is received, a copy of the client SYN segment is sent to the server and the TCP three-way handshake is performed between the ASASM and the server. If this three-way handshake completes, the connection may resume as normal. If the client does not respond during any part of the connection phase, then the ASASM retransmits the necessary segment using exponential back-offs.

TDP
Tag Distribution Protocol. TDP is used by tag switching devices to distribute, request, and release tag binding information for multiple network layer protocols in a tag switching network. TDP does not replace routing protocols. Instead, it uses information learned from routing protocols to create tag bindings. TDP is also used to open, monitor, and close TDP sessions and to indicate errors that occur during those sessions. TDP operates over a connection-oriented transport layer protocol with guaranteed sequential delivery (such as TCP). The use of TDP does not preclude the use of other mechanisms to distribute tag binding information, such as piggybacking information on other protocols.

Telnet
A terminal emulation protocol for TCP/IP networks such as the Internet. Telnet is a common way to control web servers remotely; however, its security vulnerabilities have led to its replacement by SSH.

TFTP
Trivial File Transfer Protocol. TFTP is a simple protocol used to transfer files. It runs on UDP and is explained in depth in RFC 1350.

TID
Tunnel Identifier.

TLS
Transport Layer Security. A future IETF protocol to replace SSL.

traffic policing
The traffic policing feature ensures that no traffic exceeds the maximum rate (bits per second) that you configure, which ensures that no one traffic flow can take over the entire resource.

transform set
See IPsec transform set.

translate, translation
See xlate.

transparent firewall mode
A mode in which the ASASM is not a router hop. You can use transparent firewall mode to simplify your network configuration or to make the ASASM invisible to attackers. You can also use transparent firewall mode to allow traffic through that would otherwise be blocked in routed firewall mode. See also routed firewall mode.

transport mode
An IPsec encryption mode that encrypts only the data portion (payload) of each packet but leaves the header untouched. Transport mode is less secure than tunnel mode.

TSP
TAPI Service Provider. See also TAPI.

tunnel mode
An IPsec encryption mode that encrypts both the header and data portion (payload) of each packet. Tunnel mode is more secure than transport mode.
tunnel
A method of transporting data in one protocol by encapsulating it in another protocol. Tunneling is used for reasons of incompatibility, implementation simplification, or security. For example, a tunnel lets a remote VPN client have encrypted access to a private network.

Turbo ACL
Increases ACL lookup speeds by compiling them into a set of lookup tables. Packet headers are used to access the tables in a small, fixed number of lookups, independent of the existing number of ACL entries.

U
UDP
User Datagram Protocol. A connectionless transport layer protocol in the IP protocol stack. UDP is a simple protocol that exchanges datagrams without acknowledgments or guaranteed delivery, which requires other protocols to handle error processing and retransmission. UDP is defined in RFC 768.

UMTS
Universal Mobile Telecommunication System. An extension of GPRS networks that moves toward an all-IP network by delivering broadband information, including commerce and entertainment services, to mobile users via fixed, wireless, and satellite networks.

Unicast RPF
Unicast Reverse Path Forwarding. Unicast RPF guards against spoofing by ensuring that packets have a source IP address that matches the correct source interface according to the routing table.

URL

user EXEC mode
The lowest privilege level at the ASA CLI. The user EXEC mode prompt appears as follows when you first access the ASASM:

hostname>

See also command-specific configuration mode, global configuration mode, and privileged EXEC mode.

UTC
Coordinated Universal Time. The time zone at zero degrees longitude, previously called Greenwich Mean Time (GMT) and Zulu time. UTC replaced GMT in 1967 as the world time standard. UTC is based on an atomic time scale rather than an astronomical time scale.

UTRAN
Universal Terrestrial Radio Access Network. Networking protocol used for implementing wireless networks in UMTS. GTP allows multi-protocol packets to be tunneled through a UMTS/GPRS backbone between a GGSN, an SGSN and the UTRAN.

UUIE
User-User Information Element. An element of an H.225 packet that identifies the users implicated in the message.

V
VLAN
Virtual LAN. A group of devices on one or more LANs that are configured (using management software) so that they can communicate as if they were attached to the same physical network cable, when they are located on a number of different LAN segments. Because VLANs are based on logical instead of physical connections, they are extremely flexible.
VoIP

Voice over IP. VoIP carries normal voice traffic, such as telephone calls and faxes, over an IP-based network. DSP segments the voice signal into frames, which are coupled in groups of two and stored in voice packets. These voice packets are transported using IP in compliance with ITU-T specification H.323.

VPN

Virtual Private Network. A network connection between two peers over the public network that is made private by strict authentication of users and the encryption of all data traffic. You can establish VPNs between clients, such as PCs, or a headend, such as the ASASM.

virtual firewall

See security context.

VSA

Vendor-specific attribute. An attribute in a RADIUS packet that is defined by a vendor rather than by RADIUS RFCs. The RADIUS protocol uses IANA-assigned vendor numbers to help identify VSAs. This lets different vendors have VSAs of the same number. The combination of a vendor number and a VSA number makes a VSA unique. For example, the cisco-av-pair VSA is attribute 1 in the set of VSAs related to vendor number 9. Each vendor can define up to 256 VSAs. A RADIUS packet contains any VSAs attribute 26, named Vendor-specific. VSAs are sometimes referred to as subattributes.

W

WAN

wide-area network. Data communications network that serves users across a broad geographic area and often uses transmission devices provided by common carriers.

WCCP

Web Cache Communication Protocol. Transparently redirects selected types of traffic to a group of web cache engines to optimize resource usage and lower response times.

Websense

A content filtering solution that manages employee access to the Internet. Websense uses a policy engine and a URL database to control user access to websites.

WEP

Wired Equivalent Privacy. A security protocol for wireless LANs, defined in the IEEE 802.11b standard.

WINS

Windows Internet Naming Service. A Windows system that determines the IP address associated with a particular network device, also known as name resolution. WINS uses a distributed database that is automatically updated with the NetBIOS names of network devices currently available and the IP address assigned to each one. WINS provides a distributed database for registering and querying dynamic NetBIOS names to IP address mapping in a routed network environment. It is the best choice for NetBIOS name resolution in such a routed network because it is designed to solve the problems that occur with name resolution in complex networks.

X

X.509

A widely used standard for defining digital certificates. X.509 is actually an ITU recommendation, which means that it has not yet been officially defined or approved for standardized usage.
xauth  See IKE Extended Authentication.

xlate  An xlate, also referred to as a translation entry, represents the mapping of one IP address to another, or the mapping of one IP address/port pair to another.
Symbols

/bits subnet masks  B-3
?
command string  A-4
help  A-4

A

AAA

about  33-1
accounting  35-18
authentication
  CLI access  34-18
  network access  35-2
  privileged EXEC mode  34-19
authorization
  command  34-21
  downloadable access lists  35-14
  network access  35-11
local database support  33-7
performance  35-1
server  52-4
  adding  33-11
  types  33-1
support summary  33-3
web clients  35-6
abbreviating commands  A-3

ABR
  definition of  22-2
Access Group pane
  description  24-7
access lists

about  13-1
ACE logging, configuring  18-1
deny flows, managing  18-5
downloadable  35-14
global access rules  32-2
implicit deny  13-3, 32-3
inbound  32-3
IP address guidelines  13-3
IPv6
  about  17-1
  configuring  17-4
  default settings  17-3
logging  18-1
NAT guidelines  13-3
object groups  12-2
outbound  32-3
remarks  14-5
scheduling activation  12-16
types  13-1

ACEs
  See access lists

activation key
  entering  4-10
  location  4-9
  obtaining  4-9

Active/Active failover
  about  51-1
  actions  51-5
  command replication  51-3
  configuration synchronization  51-3
  configuring
    asymmetric routing support  51-18
    failover criteria  51-16
failover group preemption 51-12
HTTP replication 51-14
interface monitoring 51-14
virtual MAC addresses 51-16
device initialization 51-3
duplicate MAC addresses, avoiding 51-2, 51-17
optional settings
  about 51-6
  configuring 51-12
primary status 51-2
secondary status 51-2
triggers 51-4
Active/Standby failover
  about 50-1
  actions 50-4
  command replication 50-3
  configuration synchronization 50-2
device initialization 50-2
  primary unit 50-2
  secondary unit 50-2
  triggers 50-4
Active Directory procedures C-16 to ??
ActiveX filtering 36-2
Adaptive Security Algorithm 1-10
Add/Edit Access Group dialog box
  description 24-7
Add/Edit IGMP Join Group dialog box
  description 24-6
Add/Edit OSPF Neighbor Entry dialog box 22-12
admin context
  about 6-2
  changing 6-23
administrative access
  using ICMP for 34-11
administrative distance 20-3, 20-5
AIP SSM
  port-forwarding
    enabling 7-4, 8-6
  alternate address, ICMP message 8-15
analyzing syslog messages 52-2
application inspection
  about 39-1
  applying 39-6
  configuring 39-6
  inspection class map 31-6
  inspection policy map 31-2
  security level requirements 7-2, 8-2
  special actions 31-1
area border router 22-2
ARP
  NAT 27-22
ARP inspection
  about 5-10
  enabling 5-12
  static entry 5-11
ARP spoofing 5-10
ARP test, failover 49-17
ASA (Adaptive Security Algorithm) 1-10
ASBR
definition of 22-2
ASDM software
  allowing access 34-6
  installing 56-2
ASR 51-18
asymmetric routing
  TCP state bypass 44-4
asymmetric routing support 51-18
attacks
  DNS request for all records 48-10
  DNS zone transfer 48-10
  DNS zone transfer from high port 48-10
  fragmented ICMP traffic 48-6, 48-9
  IP fragment 48-4, 48-7
  IP impossible packet 48-4, 48-7
  large ICMP traffic 48-6, 48-9
  ping of death 48-6, 48-9
  proxied RPC request 48-10
  statd buffer overflow 48-11
TCP NULL flags 48-6, 48-9
TCP SYN+FIN flags 48-6, 48-9
attributes
   RADIUS C-27
attribute-value pairs
   TACACS+ C-38
authentication
about 33-2
CLI access 34-18
FTP 35-3
HTTP 35-3
network access 35-2
privileged EXEC mode 34-19
Telnet 35-3
web clients 35-6
authorization
about 33-2
command 34-21
downloadable access lists 35-14
network access 35-11
autostate messaging 2-10
Auto-Update, configuring 56-16

B
Baltimore Technologies, CA server support 38-4
basic threat detection
   See threat detection
bits subnet masks B-3
Botnet Traffic Filter
   actions 46-2
   address categories 46-2
blacklist
   adding entries 46-9
   description 46-2
blocking traffic manually 46-15
classifying traffic 46-12
configuring 46-6
databases 46-2
default settings 46-6
DNS Reverse Lookup Cache
   information about 46-4
   maximum entries 46-4
   using with dynamic database 46-10
DNS snooping 46-10
dropping traffic 46-13
   graylist 46-13
dynamic database
   enabling use of 46-7
   files 46-3
   information about 46-2
   searching 46-16
   updates 46-7
examples 46-19
feature history 46-22
   graylist
      description 46-2
      dropping traffic 46-13
guidelines and limitations 46-6
information about 46-1
licensing 46-6
monitoring 46-17
static database
   adding entries 46-9
   information about 46-3
syslog messages 46-17
task flow 46-7
threat level
   dropping traffic 46-13
whitelist
   adding entries 46-9
   description 46-2
working overview 46-5
BPDUs
   forwarding on the switch 2-10
bridge
   entry timeout 5-15
table, See MAC address table
broadcast Ping test 49-17
building blocks 12-1
bypassing firewall checks 44-3
bypassing the firewall, in the switch 2-7

C
CA
CRs and 38-2
public key cryptography 38-2
revoked certificates 38-2
supported servers 38-4
capturing packets 57-14
CA server
  Digicert 38-4
  Geotrust 38-4
  Godaddy 38-4
  iPlanet 38-4
  Netscape 38-4
  RSA Keon 38-4
  Thawte 38-4
Catalyst 6500
  See switch
certificate
  enrollment protocol 38-11
Certificate Revocation Lists
  See CRLs
change query interval 24-8
change query response time 24-8
change query timeout value 24-8
changing between contexts 6-22
changing the severity level 52-18
Cisco 7600
  See switch
Cisco-AV-Pair LDAP attributes C-13
Cisco IOS CS CA
  server support 38-4
Cisco IP Phones
  DHCP 10-6
  Cisco IP Phones, application inspection 41-25
Class A, B, and C addresses B-1
class-default class map 30-9
classes, logging
  filtering messages by 52-16
  message class variables 52-4
types 52-4
classes, resource
  See resource management
class map
  inspection 31-6
Layer 3/4
  management traffic 30-14
  match commands 30-12, 30-15
  through traffic 30-12
  regular expression 12-15
CLI
  abbreviating commands A-3
  adding comments A-5
  command line editing A-3
  command output paging A-5
displaying A-5
  help A-4
  paging A-5
  syntax formatting A-3
command authorization
  about 34-14
  configuring 34-21
  multiple contexts 34-15
command prompts A-2
comments
  configuration A-5
configuration
  clearing 3-15
  comments A-5
  saving 3-12
  switch 2-1
text file 3-15
  URL for a context 6-20
Index

viewing 3-14
configuration examples
  logging 52-20
configuration examples for SNMP 54-27
configuration mode
  accessing 3-3
  prompt A-2
connection blocking 48-2
connection limits
  configuring 44-1
  per context 6-16
console port logging 52-11
context mode 25-2
context modes 20-2, 21-3, 22-3, 23-3, 24-3
contexts
  See security contexts
conversion error, ICMP message B-15
copying files using copy smb command 56-8
Coredump 57-14
crash dump 57-14
creating a custom event list 52-13
custom messages list
  logging output destination 52-4
cut-through proxy
  AAA performance 35-1
data flow
  routed firewall 5-17
  transparent firewall 5-23
date and time in messages 52-18
DDNS 11-2
debug messages 57-13
default
  class 6-9
  routes, defining equal cost routes 20-4
default policy 30-7
default routes
  about 20-4
  configuring 20-4
delay sending flow-create events
  flow-create events
    delay sending 53-9
deleting files from Flash 56-2
deny flows, logging 18-5
device ID, including in messages 52-17
device ID in messages 52-17
DHCP
  Cisco IP Phones 10-6
    options 10-4
    relay 10-7
    server 10-3
    transparent firewall 32-5
DHCP Relay panel 11-6
DHCP services 9-3
DiffServ preservation 45-2
directory hierarchy search C-3
disabling messages 52-18
disabling messages, specific message IDs 52-18
DMZ, definition 1-7
DNS
  dynamic 11-2
  inspection
    about 40-2
    managing 40-1
    rewrite, about 40-2
    rewrite, configuring 40-3
    NAT effect on 27-24
    server, configuring 9-8
DNS request for all records attack 48-10
DNS zone transfer attack 48-10
DNS zone transfer from high port attack 48-10
domain name 9-3
dotted decimal subnet masks B-3
downloadable access lists
  configuring 35-14
### Index

- **converting netmask expressions** 35-18
- **DSCP preservation** 45-2
- **dual IP stack, configuring** 7-2
- **dual-ISP support** 20-6
- **Dynamic DNS** 11-2
- **dynamic NAT**
  - about 27-8
  - network object NAT 28-4
  - twice NAT 29-4
- **dynamic PAT**
  - network object NAT 28-6
  - *See also* NAT
  - twice NAT 29-8

### E

- **echo reply, ICMP message** B-15
- **ECMP** 20-3
- **editing command lines** A-3
- **EIGRP** 32-5
  - DUAL algorithm 25-2
  - hello interval 25-13
  - hello packets 25-1
  - hold time 25-2, 25-13
  - neighbor discovery 25-1
  - stub routing 25-3
  - stuck-in-active 25-2
- **enabling logging** 52-6
- **enabling secure logging** 52-16
- **Enterprises** 10-6
- **Entrust, CA server support** 38-4
- **established command, security level requirements** 7-2, 8-2
- **Ethernet**
  - MTU 7-8, 8-9
- **EtherType access list**
  - compatibility with extended access lists 32-2
  - implicit deny 32-3
- **evaluation license** 4-4
- **exporting NetFlow records** 53-5

### F

- **extended ACLs**
  - configuring for management traffic 14-2
- **facility, syslog** 52-9
- **failover**
  - about 49-1
  - *Active/Active, See* Active/Active failover
  - *Active/Standby, See* Active/Standby failover
  - configuration file
    - terminal messages, Active/Active 51-3
    - terminal messages, Active/Standby 50-2
  - contexts 50-2
  - debug messages 49-19
  - disabling 50-17, 51-24
  - Ethernet failover cable 49-3
  - failover link 49-2
  - forcing 50-16, 51-24
  - guidelines 54-16
  - health monitoring 49-16
  - interface health 49-17
  - interface monitoring 49-17
  - interface tests 49-17
  - link communications 49-2
  - MAC addresses
    - about 50-2
    - automatically assigning 6-12
  - module placement
    - inter-chassis 49-10
    - intra-chassis 49-10
  - monitoring, health 49-16
  - network tests 49-17
  - primary unit 50-2
  - restoring a failed group 50-17, 51-24
  - restoring a failed unit 50-17, 51-24
  - secondary unit 50-2
  - SNMP syslog traps 49-19
Stateful Failover, See Stateful Failover
state link 49-3
switch configuration 2-9
system log messages 49-18
system requirements 49-2
testing 50-17, 51-24
trunk 2-10
type selection 49-7
unit health 49-17
fast path 1-11
Fibre Channel interfaces
default settings 15-2, 16-2, 32-7
filtering
ActiveX 36-2
FTP 36-14
Java applet 36-4
Java applets 36-4
security level requirements 7-2, 8-2
servers supported 36-6
show command output A-4
URLs 36-1, 36-7
filtering messages 52-4
firewall mode
about 5-1
configuring 5-1
Flash memory
removing files 56-2
flash memory available for logs 52-15
flow-export actions 53-4
format of messages 52-3
fragmented ICMP traffic attack 48-6, 48-9
fragment protection 1-8
fragment size 48-2
FTP inspection
about 40-11
configuring 40-11
G
generating RSA keys 38-9
groups
SNMP 54-15
GTP inspection
about 43-3
configuring 43-3
H
H.225 timeouts 41-9
H.245 troubleshooting 41-10
H.323
transparent firewall guidelines 5-4
H.323 inspection
about 41-4
configuring 41-3
limitations 41-5
troubleshooting 41-10
help, command line A-4
high availability
about 49-1
host
SNMP 54-15
hostname
configuring 9-2
in banners 9-2
multiple context mode 9-2
hosts, subnet masks for B-3
HSRP 5-3
HTTP
filtering 36-1
HTTP(S)
authentication 34-18
filtering 36-7
HTTP inspection
about 40-16
configuring 40-16
HTTPS/Telnet/SSH
- allowing network or host access to ASDM 34-1

ICMP
- rules for access to ADSM 34-11
- testing connectivity 57-1
- type numbers B-15

identity NAT
- about 27-11
- network object NAT 28-12
- twice NAT 29-18

ILS inspection 42-1

IM 41-19

implementing SNMP 54-15

inbound access lists 32-3

information reply, ICMP message B-15

information request, ICMP message B-15

inside, definition 1-7

inspection_default class-map 30-8

inspection engines
- See application inspection installation
- module verification 2-3

Instant Messaging inspection 41-19

interface
- MTU 7-8, 8-9

interfaces
- default settings 15-2, 16-2, 32-7
- failover monitoring 49-17
- IP address 7-5

MAC addresses
- automatically assigning 6-21
- manually assigning to interfaces 7-8, 8-9
- mapped name 6-19

naming, physical and subinterface 7-5, 8-7

turning off 7-14, 8-13

turning on 7-14, 8-13

IOS
- upgrading 2-1

IP addresses
- classes B-1
- interface 7-5
- management, transparent firewall 8-5
- private B-2
- subnet mask B-4

IP fragment attack 48-4, 48-7

IP impossible packet attack 48-4, 48-7

IP overlapping fragments attack 48-5

IP spoofing, preventing 48-1

IP teardrop attack 48-5

IPv6
- commands 19-10
- configuring alongside IPv4 7-2
- default route 20-5
- dual IP stack 7-2
- duplicate address detection 7-9, 8-10
- neighbor discovery 26-1
- router advertisement messages 26-3
- static neighbors 26-4
- static routes 20-5

IPv6 addresses
- anycast B-9
- command support for 19-10
- format B-5
- multicast B-8
- prefixes B-10
- required B-10
- types of B-6
- unicast B-6

IPv6 prefixes 26-11

IPX 2-7

Java applet filtering 36-4

Java applets, filtering 36-2
Join Group pane
  description 24-6
jumbo frames 7-7, 8-8

K
Kerberos
  configuring 33-11
  support 33-6

L
large ICMP traffic attack 48-6, 48-9
latency
  about 45-1
Layer 2 firewall
  See transparent firewall
Layer 2 forwarding table
  See MAC address table
Layer 3/4
  matching multiple policy maps 30-5
LDAP
  application inspection 42-1
  attribute mapping 33-16, 33-17
  Cisco-AV-pair C-13
  configuring 33-11
  configuring a AAA server C-2 to ??
  directory search C-3
  example configuration procedures C-16 to ??
  hierarchy example C-3
SASL 33-6
  user authentication 33-6
licenses
  activation key
    entering 4-10
    location 4-9
    obtaining 4-9
ASA 5580 4-2
default 4-3
evaluation 4-4
failover 4-8
guidelines 4-8
managing 4-1
preinstalled 4-3
Product Authorization Key 4-9
temporary 4-4
viewing current 4-11
VPN Flex 4-4
licensing requirements
  logging 52-5
licensing requirements for SNMP 54-16
link up/down test 49-17
local user database
  adding a user 33-19
  configuring 33-19
  logging in 34-19
  support 33-7
lockout recovery 34-30
logging
  access lists 18-1
  classes
    filtering messages by 52-4
    types 52-4, 52-16
device-id, including in system log messages 52-17
e-mail
  source address 52-10
EMBLEM format 52-14
facility option 52-9
filtering
  by message class 52-16
  by message list 52-4
  by severity level 52-1
logging queue, configuring 52-15
output destinations 52-8
  console port 52-8, 52-10, 52-11
  internal buffer 52-1, 52-6
  Telnet or SSH session 52-6
queue
  changing the size of 52-15
  configuring 52-15
  viewing queue statistics 52-19
severity level, changing 52-19
timestamp, including 52-18
logging feature history 52-20
logging queue
  configuring 52-15
login
  banner, configuring 34-7
  FTP 35-3
local user 34-19
password 9-1
session 3-3
SSH 3-3, 34-5
Telnet 3-3, 9-1
loops, avoiding 2-10

M

MAC addresses
  automatically assigning 6-21
  failover 50-2
  manually assigning to interfaces 7-8, 8-9
  security context classification 6-3
MAC address table
  about 5-23
  built-in-switch 5-14
  entry timeout 5-15
  MAC learning, disabling 5-16
  resource management 6-16
  static entry 5-15
MAC learning, disabling 5-16
management interfaces
  default settings 15-2, 16-2, 32-7
management IP address, transparent firewall 8-5
man-in-the-middle attack 5-10
mapped addresses
  guidelines 27-21
mapped interface name 6-19
mask
  reply, ICMP message B-15
  request, ICMP message B-15
Master Passphrase 9-3
match commands
  inspection class map 31-4
  Layer 3/4 class map 30-12, 30-15
message filtering 52-4
message list
  filtering by 52-4
message-of-the-day banner 34-8
messages, logging
  classes
    about 52-4
    list of 52-4
  component descriptions 52-3
  filtering by message list 52-4
  format of 52-3
  message list, creating 52-13
  severity levels 52-3
messages classes 52-4
messages in EMBLEM format 52-14
metacharacters, regular expression 12-13
MGCP inspection
  about 41-11
  configuring 41-11
mgmt0 interfaces
  default settings 15-2, 16-2, 32-7
MIBs 54-3
MIBs for SNMP 54-28
Microsoft Windows CA, supported 38-4
mobile redirection, ICMP message B-15
mode
  context 6-14
  firewall 5-1
modular policy framework
  configuring flow-export actions for NetFlow 53-5
monitoring
  failover  49-16
  OSPF  22-16
  resource management  6-27
  SNMP  54-1
monitoring logging  52-19
monitoring NSEL  53-10
More prompt  A-5
MPF
  default policy  30-7
  examples  30-18
  feature directionality  30-3
  features  30-2
  flows  30-5
  matching multiple policy maps  30-5
  service policy, applying  30-17
  See also class map
  See also policy map
MPLS
  LDP  32-6
  router-id  32-6
  TDP  32-6
MRoute pane
  description  24-4
MSFC
  overview  1-5
  SVIs  2-7
MTU  7-8, 8-9
multicast traffic  5-3
multiple context mode
  logging  52-2
  See security contexts
multiple SVIs  2-6

N
naming an interface
  other models  7-5, 8-7
NAT
  about  27-1
  bidirectional initiation  27-2
  disabling proxy ARP for global addresses  19-11
  DNS  27-24
  dynamic
    about  27-8
    dynamic NAT
      network object NAT  28-4
      twice NAT  29-4
    dynamic PAT
      about  27-10
      network object NAT  28-6
      twice NAT  29-8
  identity
    about  27-11
    identity NAT
      network object NAT  28-12
      twice NAT  29-18
  implementation  27-16
  interfaces  27-21
  mapped address guidelines  27-21
  network object
    comparison with twice NAT  27-16
  network object NAT
    about  27-17
    configuring  28-1
    dynamic NAT  28-4
    dynamic PAT  28-6
    examples  28-15
    guidelines  28-2
    identity NAT  28-12
    monitoring  28-14
    prerequisites  28-2
    static NAT  28-9
  no proxy ARP  28-13, 29-17
  routed mode  27-13
  route lookup  28-13, 29-22
  RPC not supported with  42-4
  rule order  27-20
static
  about  27-3
  few-to-many mapping  27-7
  many-to-few mapping  27-6, 27-7
  one-to-many  27-6
static NAT
  network object NAT  28-9
  twice NAT  29-13
static with port translation
  about  27-4
terminology  27-2
transparent mode  27-13
twice NAT
  about  27-17
  comparison with network object NAT  27-16
  configuring  29-1
dynamic NAT  29-4
dynamic PAT  29-8
examples  29-22
  guidelines  29-2
  identity NAT  29-18
  monitoring  29-22
  prerequisites  29-2
static NAT  29-13
types  27-3
VPN  27-14
  VPN client rules  27-20
neighbor reachable time  26-3
neighbor solicitation messages  26-2
neighbor advertisement messages  26-2
NetFlow
  overview  53-1
NetFlow collector
  configuring  53-5
NetFlow event
  matching to configured collectors  53-5
NetFlow event logging
  disabling  53-9
Network Activity test  49-17

network object NAT
  about  27-17
  comparison with twice NAT  27-16
  configuring  28-1
dynamic NAT  28-4
dynamic PAT  28-6
examples  28-15
  guidelines  28-2
  identity NAT  28-12
  monitoring  28-14
  prerequisites  28-2
static NAT  28-9

No Payload Encryption  4-7
no proxy ARP  29-17
NSEL and syslog messages
  redundant messages  53-2
NSEL configuration examples  53-12
NSEL feature history  53-14
NSEL licensing requirements  53-3
NSEL runtime counters
  clearing  53-10
NTLM support  33-6
NT server
  configuring  33-11
  support  33-6

O

object groups
  about  12-1
  configuring  12-6
  removing  12-11
object NAT
  See network object NAT
open ports  B-14
OSPF
  area authentication  22-11
  area MD5 authentication  22-11
  area parameters  22-10
authentication key 22-9
authentication support 22-2
cost 22-9
dead interval 22-9
defining a static neighbor 22-12
interaction with NAT 22-2
interface parameters 22-8
link-state advertisement 22-2
logging neighbor states 22-13
LSAs 22-2
MD5 authentication 22-9
monitoring 22-16
NSSA 22-11
packet pacing 22-16
processes 22-2
redistributing routes 22-4
route calculation timers 22-13
route summarization 22-7
outbound access lists 32-3
output destination 52-5
output destinations 52-1, 52-6
e-mail address 52-1, 52-6
SNMP management station 52-1, 52-6
Telnet or SSH session 52-1, 52-6
outside, definition 1-7
oversubscribing resources 6-8

P

packet
  capture 57-14
  classifier 6-3
packet flow
  routed firewall 5-17
  transparent firewall 5-23
packet trace, enabling 57-7
paging screen displays A-5
parameter problem, ICMP message B-15
password
  resetting on SSM hardware module 57-11
  passwords
    changing 9-2
    recovery 57-8
    security appliance 9-1
PAT
  See dynamic PAT
ping
  See ICMP
ping of death attack 48-6, 48-9
PKI protocol 38-11
policy, QoS 45-1
policy map
  inspection 31-2
  Layer 3/4
    about 30-1
    feature directionality 30-3
    flows 30-5
pools, address
  DHCP 10-3
port-forwarding
  enabling 7-4, 8-6
ports
  open on device B-14
TCP and UDP B-11
port translation
  about 27-4
primary unit, failover 50-2
private networks B-2
privileged EXEC mode
  accessing 3-3
privileged mode
  prompt A-2
Product Authorization Key 4-9
prompts
  command A-2
  more A-5
protocol numbers and literal values B-11
Protocol pane (PIM)
description 24-10
proxied RPC request attack 48-10
proxy ARP
  NAT
    proxy ARP 27-22
proxy ARP, disabling 19-11
proxy servers
  SIP and 41-19
public key cryptography 38-2

QoS
  about 45-1, 45-2
  DiffServ preservation 45-2
  DSCP preservation 45-2
  policies 45-1
  statistics 45-6
  token bucket 45-2
  viewing statistics 45-6
Quality of Service
  See QoS

question mark
  command string A-4
  help A-4
queue, logging
  changing the size of 52-15
  viewing statistics 52-19

RADIUS
  attributes C-27
  Cisco AV pair C-13
  configuring a AAA server C-27
  configuring a server 33-11
  downloadable access lists 35-14
  network access authentication 35-4
  network access authorization 35-14
  support 33-3
  rapid link failure detection 2-10
  RAS, H.323 troubleshooting 41-10
  rate limit 52-19
  rate limiting 45-2
  RealPlayer 41-15
  redirect, ICMP message B-15
  Registration Authority description 38-2
  regular expression 12-12
  reloading
    context 6-25
    security appliance 57-8
  Request Filter pane
    description 24-11
  resetting the services module 2-11
  resetting the SSM hardware module password 57-11
  resource management
    about 6-8
    assigning a context 6-20
    class 6-15
    configuring 6-8
    default class 6-9
    monitoring 6-27
    oversubscribing 6-8
    resource types 6-16
    unlimited 6-9
  resource usage 6-30
  revoked certificates 38-2
  RFCs for SNMP 54-28
  RIP
    authentication 23-2
    definition of 23-1
    enabling 23-4
    support for 23-2
  RIP panel
    limitations 23-3
    RIP Version 2 Notes 23-3
  routed mode
INDEX

Cisco ASA Services Module CLI Configuration Guide

about 5-1
NAT 27-13
setting 5-1
route map
definition 21-1
route maps
defining 21-4
uses 21-1
router
advertisement, ICMP message B-15
solicitation, ICMP message B-15
router advertisement messages 26-3
router advertisement transmission interval 26-8
router lifetime value 26-8
routes
about default 20-4
configuring default routes 20-4
configuring IPv6 default 20-5
configuring IPv6 static 20-5
configuring static routes 20-3
routing
other protocols 32-5
RSA
keys, generating 34-4, 38-9
RTSP inspection
about 41-15
configuring 41-15
rules
ICMP 34-10
running configuration
copying 56-8
saving 3-12

S
same security level communication
enabling 7-12, 8-13
SCCP (Skinny) inspection
about 41-25
configuration 41-25
configuring 41-25
SDI
configuring 33-11
support 33-5
secondary unit, failover 50-2
security appliance
CLI A-1
managing licenses 4-1
managing the configuration 3-11
reloading 57-8
upgrading software 56-2
viewing files in Flash memory 56-1
security contexts
about 6-1
adding 6-17
admin context
about 6-2
changing 6-23
assigning to a resource class 6-20
cascading 6-6
changing between 6-22
classifier 6-3
command authorization 34-15
configuration
URL, changing 6-23
URL, setting 6-20
logging in 6-7
MAC addresses
automatically assigning 6-21
classifying using 6-3
managing 6-1, 6-22
mapped interface name 6-19
monitoring 6-26
MSFC compatibility 1-7
multiple mode, enabling 6-14
nesting or cascading 6-7
prompt A-2
reloading 6-25
removing 6-22
resource management 6-8
resource usage 6-30
saving all configurations 3-13
unsupported features 6-13
VLAN allocation 6-19
security level
about 7-1
interface 7-6, 8-7
security models for SNMP 54-15
sending messages to an e-mail address 52-10
sending messages to an SNMP server 52-12
sending messages to ASDM 52-11
sending messages to a specified output destination 52-16
sending messages to a syslog server 52-8
sending messages to a Telnet or SSH session 52-12
sending messages to the console port 52-11
sending messages to the internal log buffer 52-9
service policy
applying 30-17
default 30-17
interface 30-18
session management path 1-10
severity levels, of system log messages
changing 52-1
filtering by 52-1
list of 52-3
severity levels, of system messages
definition 52-3
show command, filtering output A-4
single mode
backing up configuration 6-15
configuration 6-14
enabling 6-14
restoring 6-15
SIP inspection
about 41-19
configuring 41-19
instant messaging 41-19
timeouts 41-24
troubleshooting 41-24
Smart Call Home monitoring 55-19
SMTP inspection 40-31
SNMP
about 54-1
failover 54-16
management station 52-1, 52-6
prerequisites 54-16
SNMP configuration 54-17
SNMP groups 54-15
SNMP hosts 54-15
SNMP monitoring 54-25, 54-26
SNMP terminology 54-2
SNMP traps 54-3
SNMP users 54-15
SNMP Version 3 54-14, 54-22
SNMP Versions 1 and 2c 54-21
source quench, ICMP message B-15
SPAN session 2-2
SSH
authentication 34-18
concurrent connections 34-2
login 34-5
password 9-1
RSA key 34-4
username 34-5
startup configuration
copying 56-8
saving 3-12
statd buffer overflow attack 48-11
Stateful Failover
about 49-8
state information 49-9
state link 49-3
stateful inspection 1-10
bypassing 44-3
state information 49-9
state link 49-3
static ARP entry 5-11
static bridge entry 5-15
Static Group pane
description 24-6
static NAT
about 27-3
few-to-many mapping 27-7
many-to-few mapping 27-6, 27-7
network object NAT 28-9
twice NAT 29-13
static NAT with port translation
about 27-4
static routes
configuring 20-3
statistics, QoS 45-6
stealth firewall
See transparent firewall
stuck-in-active 25-2
subcommand mode prompt A-2
subnet masks
/bits B-3
about B-2
address range B-4
determining B-3
dotted decimal B-3
number of hosts B-3
Sun RPC inspection
about 42-3
configuring 42-3
SVIs
configuring 2-8
multiple 2-6
overview 2-6
switch
assigning VLANs to module 2-4
autostate messaging 2-10
BPDU forwarding 2-10
configuration 2-1
failover compatibility with transparent firewall 2-10
failover configuration 2-9
trunk for failover 2-10
verifying module installation 2-3
switched virtual interfaces
See SVIs
switch MAC address table 5-14
SYN attacks, monitoring 6-31
SYN cookies 6-31
syntax formatting A-3
syslogd server program 52-5
syslog messages
analyzing 52-2
syslog messaging for SNMP 54-26
syslog server
designating more than one as output destination 52-5
EMBLEM format
configuring 52-14
enabling 52-8, 52-14
system configuration 6-2
system log messages
classes 52-4
classes of 52-4
configuring in groups
by message list 52-4
by severity level 52-1
device ID, including 52-17
disabling logging of 52-1
filtering by message class 52-4
managing in groups
by message class 52-16
output destinations 52-1, 52-6
syslog message server 52-6
Telnet or SSH session 52-6
severity levels
about 52-3
changing the severity level of a message 52-1
timestamp, including 52-18
TACACS+

command authorization, configuring 34-28
configuring a server 33-11
network access authorization 35-11
support 33-5

TCP

connection limits per context 6-16
ports and literal values B-11
sequence number randomization
   disabling using Modular Policy Framework 44-12

TCP Intercept
   enabling using Modular Policy Framework 44-12
monitoring 6-31

TCP normalization 44-3

TCP NULL flags attack 48-6, 48-9

TCP state bypass

AAA 44-5
   configuring 44-10
   failover 44-5
   firewall mode 44-5
   inspection 44-5
   multiple context mode 44-5
NAT 44-5
SSMs and SSCs 44-5
TCP Intercept 44-5
TCP normalization 44-5
unsupported features 44-5

TCP SYN+FIN flags attack 48-6, 48-9

Telnet
   allowing management access 34-1
   authentication 34-18
   concurrent connections 34-2
   login 34-4
   password 9-1

   template timeout intervals
      configuring for flow-export actions 53-7

   temporary license 4-4
   testing configuration 57-1
   threat detection
      basic
         drop types 47-2
         enabling 47-4
         overview 47-2
         rate intervals 47-2
         rate intervals, setting 47-4
         statistics, viewing 47-5
         system performance 47-3
scanning
         attackers, viewing 47-18
         default limits, changing 47-17
         enabling 47-17
         host database 47-15
         overview 47-15
         shunned hosts, releasing 47-18
         shunned hosts, viewing 47-17
         shunning attackers 47-17
         system performance 47-15
         targets, viewing 47-18
scanning statistics
   enabling 47-7
   system performance 47-6
   viewing 47-9

time exceeded, ICMP message B-15
time ranges, access lists 12-16
timestamp, including in system log messages 52-18
timestamp reply, ICMP message B-15
timestamp request, ICMP message B-15
token bucket 45-2
traffic flow
   routed firewall 5-17
   transparent firewall 5-23

   transparent firewall about 5-2
   ARP inspection about 5-10
enabling 5-12
static entry 5-11
data flow 5-23
DHCP packets, allowing 32-5
guidelines 5-7
H.323 guidelines 5-4
HSRP 5-3
MAC address timeout 5-15
MAC learning, disabling 5-16
management IP address 8-5
multicast traffic 5-3
packet handling 32-5
static bridge entry 5-15
unsupported features 5-7
VRRP 5-3
transparent mode
NAT 27-13
troubleshooting
H.323 41-9
H.323 RAS 41-10
SIP 41-24
troubleshooting SNMP 54-23
Trusted Flow Acceleration
modes 5-6, 5-11, 5-14, 14-1, 32-7, 51-7
trustpoint 38-3
twice NAT
about 27-17
comparison with network object NAT 27-16
configuring 29-1
dynamic NAT 29-4
dynamic PAT 29-8
dynamic PAT examples 29-22
guidelines 29-2
identity NAT 29-18
monitoring 29-22
prerequisites 29-2
static NAT 29-13

U

UDP
connection limits per context 6-16
connection state information 1-11
ports and literal values B-11
unprivileged mode
accessing 3-3
unreachable, ICMP message B-15
unreachable messages
required for MTU discovery 34-10
upgrading
IOS 2-1
URLs
context configuration, changing 6-23
collection configuration, setting 6-20
filtering 36-1
filtering, about 36-7
filtering, configuration 36-11
user EXEC mode
prompt A-2
username
adding 33-19
encrypted 33-21
password 33-21
users
SNMP 54-15

V

VeriSign, configuring CAs example 38-4
viewing QoS statistics 45-6
viewing RMS 56-19
virtual firewalls
See security contexts
virtual HTTP 35-3
virtual reassembly 1-8
VLANs
allocating to a context 6-19
assigning to FWSM  2-4
interfaces  2-4
mapped interface name  6-19
VoIP
proxy servers  41-19
troubleshooting  41-9
VPN
address range, subnets  B-4
VPN client
NAT rules  27-20
VPN flex license  4-4
VRRP  5-3

W

WCCP  37-1
web caching  37-1
web clients, secure authentication  35-6