Cisco ASA 1000V CLI Configuration Guide for ASDM Mode

Software Version 8.7 for the ASA 1000V
PART 1

CHAPTER 1

Introduction to the Cisco ASA 1000V

New Features in ASA 1000V Version 8.7(1) 1-1
Supported and Unsupported Features of the Cisco ASA 1000V 1-3
VMware Feature Support for the ASA 1000V 1-5
Task Flow for Configuring the ASA 1000V 1-5
Cloning the ASA 1000V 1-6
  Requirements for Cloning 1-6
  Preparing the Master ASA 1000V 1-6
  Cloning the Master 1-6
Licensing Enforcement for the ASA 1000V 1-7
Configuration Examples for the ASA 1000V 1-8
Monitoring the ASA 1000V 1-8
Firewall Functional Overview 1-8
  Security Policy Overview 1-8
    Permitting or Denying Traffic with Access Lists (Rules) 1-9
    Applying NAT 1-9
    Protection from IP Fragments 1-9
    Applying Application Inspection 1-9
    Applying Connection Limits and TCP Normalization 1-9
  Failover Overview 1-9
  Firewall Mode Overview 1-10
  Stateful Inspection Overview 1-10
IPsec Site-to-Site VPN Functional Overview 1-11
Additional References 1-11
CHAPTER 2

Getting Started 2-1
Deploying and Powering Up the ASA 1000V 2-1
Default Configuration after Deployment 2-1
Accessing the ASA 1000V Command-Line Interface 2-2
Starting ASDM 2-3
   Connecting to ASDM for the First Time 2-4
   Starting ASDM from the ASDM-IDM Launcher 2-5
   Starting ASDM from the Java Web Start Application 2-5
   Using ASDM in Demo Mode 2-6
ASA 1000V File Storage 2-7
Working with the Configuration 2-7
   Saving Configuration Changes 2-7
   Copying the Startup Configuration to the Running Configuration 2-8
   Viewing the Configuration 2-8
   Clearing and Removing Configuration Settings 2-9
   Creating Text Configuration Files Offline 2-9
   Applying Configuration Changes to Connections 2-10

CHAPTER 3

Configuring Active/Standby Failover 3-1
Introduction to Failover and High Availability 3-1
   Failover System Requirements 3-2
   Information About Active/Standby Failover 3-2
      Active/Standby Failover Overview 3-2
      Primary/Secondary Status and Active/Standby Status 3-2
      Device Initialization and Configuration Synchronization 3-3
   Command Replication 3-3
   Failover Triggers 3-4
   Failover Actions 3-5
   Optional Active/Standby Failover Settings 3-6
   Failover and Stateful Failover Links 3-6
      Failover Link 3-6
      Stateful Failover Link 3-7
      Avoiding Interrupted Failover Links 3-7
   Stateless (Regular) and Stateful Failover 3-9
      Stateless (Regular) Failover 3-9
      Stateful Failover 3-9
### Contents

Auto Update Server Support in Failover Configurations 3-10  
Auto Update Process Overview 3-10  
Monitoring the Auto Update Process 3-11  
Failover Health Monitoring 3-12  
ASA 1000V Health Monitoring 3-13  
Interface Monitoring 3-13  
Failover Messages 3-14  
Failover System Messages 3-14  
Debugging Messages 3-14  
SNMP 3-14  
Prerequisites for Active/Standby Failover 3-14  
Guidelines and Limitations 3-15  
Configuring Active/Standby Failover 3-15  
Task Flow for Configuring Active/Standby Failover 3-15  
Configuring the Primary ASA 1000V 3-16  
Configuring the Secondary ASA 1000V 3-18  
Configuring Optional Active/Standby Failover Settings 3-19  
Enabling HTTP Replication with Stateful Failover 3-19  
Disabling and Enabling Interface Monitoring 3-20  
Configuring Failover Criteria 3-20  
Configuring the ASA 1000V and Interface Health Poll Times 3-21  
Configuring Virtual MAC Addresses 3-21  
Controlling Failover 3-22  
Forcing Failover 3-23  
Disabling Failover 3-23  
Restoring a Failed ASA 1000V 3-23  
Testing the Failover Functionality 3-23  
Monitoring Active/Standby Failover 3-24  
Feature History for Active/Standby Failover 3-24

#### Configuring Interfaces 4-1

Information About Interfaces 4-1  
Ethernet Interfaces 4-1  
Security Profile Interfaces 4-2  
Security Profile Coordination with VNMC 4-2  
Security Profiles on the Inside Interface 4-2  
Interface-Based Policy 4-2  
Port Profiles for Security Profile Interfaces 4-3  
How to Apply the Security Policy to Interfaces 4-3
To-the-Box Traffic 4-3
vPath Tagging 4-3
Interface Security Levels 4-4
Guidelines and Limitations 4-4
Default Settings 4-4
Configuring Communication with VNMC 4-5
Configuring Interfaces 4-6
  Configuring the Inside and Outside Ethernet Interfaces 4-6
  Configuring Security Profile Interfaces 4-8
  Associating Security Profile Interfaces with an Ethernet Interface 4-9
  Setting the vPath MTU 4-9
Monitoring Interfaces 4-10
Feature History for Interfaces 4-11

CHAPTER 5 Configuring the Hostname, Domain Name, Passwords, and Other Basic Settings 5-1
Configuring the Hostname, Domain Name, and Passwords 5-1
  Changing the Login Password 5-2
  Changing the Enable Password 5-2
  Setting the Hostname 5-3
  Setting the Domain Name 5-3
Setting the Date and Time 5-3
  Setting the Time Zone and Daylight Saving Time Date Range 5-4
  Setting the Date and Time Using an NTP Server 5-5
  Setting the Date and Time Manually 5-6
Configuring the Master Passphrase 5-6
  Information About the Master Passphrase 5-6
  Adding or Changing the Master Passphrase 5-7
  Disabling the Master Passphrase 5-9
  Recovering the Master Passphrase 5-9
Configuring the DNS Server 5-10

CHAPTER 6 Configuring DHCP 6-1
Information About DHCP 6-1
Guidelines and Limitations 6-1
Configuring a DHCP Server 6-2
  Enabling the DHCP Server 6-2
  Configuring DHCP Options 6-3
    Options that Return an IP Address 6-3
Options that Return a Text String  6-3
Options that Return a Hexadecimal Value  6-4
Configuring DHCP Relay Services  6-5
DHCP Monitoring Commands  6-6

**CHAPTER  7**

Configuring Static and Default Routes  7-1
Information About Routing  7-1
Switching  7-1
Information About Static and Default Routes  7-2
How Routing Behaves Within the ASA 1000V  7-2
Egress Interface Selection Process  7-2
Next Hop Selection Process  7-3
Information About the Routing Table  7-3
Displaying the Routing Table  7-3
How Forwarding Decisions Are Made  7-4
Configuring Static and Default Routes  7-4
Configuring a Default Static Route  7-4
Configuring a Static Route  7-6
Adding or Editing a Static Route  7-7
Disabling Proxy ARPs  7-7
Monitoring a Static or Default Route  7-8
Configuration Examples for Static or Default Routes  7-10

**PART  3**

Configuring Objects and Access Lists

**CHAPTER  8**

Configuring Objects  8-1
Configuring Objects and Groups  8-1
Information About Objects and Groups  8-1
Information About Objects  8-2
Information About Object Groups  8-2
Guidelines and Limitations for Objects and Groups  8-2
Configuring Objects  8-3
Configuring a Network Object  8-3
Configuring a Service Object  8-3
Configuring Object Groups  8-5
Adding a Protocol Object Group  8-5
Adding a Network Object Group  8-6
Adding a Service Object Group  8-7
PART 4 Configuring Network Address Translation

CHAPTER 11 Information About NAT 11-1
Why Use NAT? 11-1
NAT Terminology 11-3
NAT Types 11-3
NAT Types Overview 11-3
Static NAT 11-4
Information About Static NAT 11-4
Information About Static NAT with Port Translation 11-4
Information About One-to-Many Static NAT 11-7
Information About Other Mapping Scenarios (Not Recommended) 11-8
Dynamic NAT 11-9
Information About Dynamic NAT 11-10
Dynamic NAT Disadvantages and Advantages 11-11
Dynamic PAT 11-11
Information About Dynamic PAT 11-11
Dynamic PAT Disadvantages and Advantages 11-12
Identity NAT 11-12
NAT for VPN 11-13
How NAT is Implemented 11-14
Main Differences Between Network Object NAT and Twice NAT 11-14
Information About Network Object NAT 11-15
Information About Twice NAT 11-15
NAT Rule Order 11-18
NAT Interfaces 11-19
Routing NAT Packets 11-20
Mapped Addresses and Routing 11-20
Determining the Egress Interface 11-22
DNS and NAT 11-22
Where to Go Next 11-24
CHAPTER 12  Configuring Network Object NAT 12-1

Information About Network Object NAT 12-1
Prerequisites for Network Object NAT 12-2
Guidelines and Limitations 12-2
Default Settings 12-2
Configuring Network Object NAT 12-3
  Configuring Dynamic NAT 12-3
  Configuring Dynamic PAT (Hide) 12-5
  Configuring Static NAT or Static NAT-with-Port-Translation 12-9
  Configuring Identity NAT 12-11
Monitoring Network Object NAT 12-13
Configuration Examples for Network Object NAT 12-13
  Providing Access to an Inside Web Server (Static NAT) 12-14
  NAT for Inside Hosts (Dynamic NAT) and NAT for an Outside Web Server (Static NAT) 12-15
  Inside Load Balancer with Multiple Mapped Addresses (Static NAT, One-to-Many) 12-16
  Single Address for FTP, HTTP, and SMTP (Static NAT-with-Port-Translation) 12-17
  DNS Server on Mapped Interface, Web Server on Real Interface (Static NAT with DNS Modification) 12-18
  DNS Server and Web Server on Mapped Interface, Web Server is Translated (Static NAT with DNS Modification) 12-20
Feature History for Network Object NAT 12-21

CHAPTER 13  Configuring Twice NAT 13-1

Information About Twice NAT 13-1
Prerequisites for Twice NAT 13-2
Guidelines and Limitations 13-2
Default Settings 13-2
Configuring Twice NAT 13-3
  Configuring Dynamic NAT 13-3
  Configuring Dynamic PAT (Hide) 13-8
  Configuring Static NAT or Static NAT-with-Port-Translation 13-15
  Configuring Identity NAT 13-20
Monitoring Twice NAT 13-24
Configuration Examples for Twice NAT 13-24
  Different Translation Depending on the Destination (Dynamic PAT) 13-24
  Different Translation Depending on the Destination Address and Port (Dynamic PAT) 13-26
Feature History for Twice NAT 13-28
PART 5
Configuring Service Policies Using the Modular Policy Framework

CHAPTER 14
Configuring a Service Policy Using the Modular Policy Framework

Information About Service Policies
Supported Features for Through Traffic
Supported Features for Management Traffic
Feature Directionality
Feature Matching Within a Service Policy
Order in Which Multiple Feature Actions are Applied
Incompatibility of Certain Feature Actions
Feature Matching for Multiple Service Policies
Guidelines and Limitations
Default Settings
Default Configuration
Default Class Maps
Task Flows for Configuring Service Policies
Task Flow for Using the Modular Policy Framework
Identifying Traffic (Layer 3/4 Class Maps)
Creating a Layer 3/4 Class Map for Through Traffic
Creating a Layer 3/4 Class Map for Management Traffic
Defining Actions (Layer 3/4 Policy Map)
Applying Actions to an Interface (Service Policy)
Monitoring Modular Policy Framework
Configuration Examples for Modular Policy Framework
Applying Inspection to HTTP Traffic
Applying Inspection to HTTP Traffic Globally
Applying Inspection and Connection Limits to HTTP Traffic to Specific Servers
Applying Inspection to HTTP Traffic with NAT
Feature History for Service Policies

CHAPTER 15
Configuring Special Actions for Application Inspections (Inspection Policy Map)

Information About Inspection Policy Maps
Guidelines and Limitations
Default Inspection Policy Maps
Defining Actions in an Inspection Policy Map
Identifying Traffic in an Inspection Class Map
Where to Go Next

Cisco ASA 1000V CLI Configuration Guide for ASDM Mode
PART 6
Configuring Access Control

CHAPTER 16
Configuring Access Rules 16-1
Information About Access Rules 16-1
How to Apply Access Rules to Interfaces 16-2
Implicit Permits 16-2
Information About Interface Access Rules and Global Access Rules 16-2
Implicit Deny 16-2
Inbound and Outbound Rules 16-3
Access Rules for Returning Traffic 16-4
Management Access Rules 16-4
Prerequisites 16-4
Default Settings 16-4
Configuring Access Rules 16-4
Monitoring Access Rules 16-5
Configuration Examples for Permitting or Denying Network Access 16-5

CHAPTER 17
Configuring Management Access 17-1
Configuring ASA 1000V Access for ASDM, Telnet, or SSH 17-1
Guidelines and Limitations 17-2
Configuring Telnet Access 17-2
Using a Telnet Client 17-3
Configuring SSH Access 17-3
Using an SSH Client 17-4
Configuring HTTPS Access for ASDM 17-5
Configuring CLI Parameters 17-5
Configuring a Login Banner 17-6
Customizing a CLI Prompt 17-7
Changing the Console Timeout 17-8
Configuring ICMP Access 17-8
Information About ICMP Access 17-8
Guidelines and Limitations 17-9
Default Settings 17-9
Configuring ICMP Access 17-9
Configuring Management Access Over an IPsec Site-to-Site Tunnel 17-10
Guidelines and Limitations 17-10
Configuring the Management Interface 17-10
Feature History for Management Access 17-10
CHAPTER 18

Configuring AAA for Management Access 18-1

Information About AAA for Management Access 18-1
  Information About Authentication 18-2
  Information About Authorization 18-2
  Information About Accounting 18-2
  Information About Management Authentication 18-2
    Comparing CLI Access with and without Authentication 18-3
    Comparing ASDM Access with and without Authentication 18-3
  Information About Command Authorization 18-3
    Supported Command Authorization Methods 18-3
    About Preserving User Credentials 18-4
Summary of Server Support 18-5
RADIUS Server Support 18-5
  Authentication Methods 18-5
  Attribute Support 18-5
TACACS+ Server Support 18-6
RSA/SDI Server Support 18-6
  RSA/SDI Version Support 18-6
  Two-step Authentication Process 18-6
  RSA/SDI Primary and Replica Servers 18-7
NT Server Support 18-7
Kerberos Server Support 18-7
LDAP Server Support 18-7
  Authentication with LDAP 18-7
  LDAP Server Types 18-8
Local Database Support, Including as a Fallback Method 18-8
  How Fallback Works with Multiple Servers in a Group 18-9
Prerequisites 18-9
Default Settings 18-10
Configuring AAA Servers and Local Users 18-10
  Task Flow for Configuring AAA Servers 18-10
Configuring AAA Server Groups 18-11
Configuring LDAP Attribute Maps 18-15
Adding a User Account to the Local Database 18-17
Authenticating Users with a Public Key for SSH 18-20
Configuring AAA for Management Access 18-20
  Configuring Authentication for CLI and ASDM Access 18-21
  Configuring Authentication to Access Privileged EXEC Mode (the enable Command) 18-21
    Configuring Authentication for the enable Command 18-22
PART 7
Configuring Application Inspection

CHAPTER 19
Getting Started with Application Layer Protocol Inspection 19-1

- Information about Application Layer Protocol Inspection 19-1
- How Inspection Engines Work 19-1
- When to Use Application Protocol Inspection 19-2

- Guidelines and Limitations 19-3
- Default Settings 19-3
- Configuring Application Layer Protocol Inspection 19-5

CHAPTER 20
Configuring Inspection of Basic Internet Protocols 20-1

- DNS Inspection 20-1
  - How DNS Application Inspection Works 20-2
  - How DNS Rewrite Works 20-2
  - Configuring DNS Rewrite 20-3
    - Configuring DNS Rewrite with Two NAT Zones 20-4
    - Overview of DNS Rewrite with Three NAT Zones 20-4
    - Configuring DNS Rewrite with Three NAT Zones 20-6
    - Configuring a DNS Inspection Policy Map for Additional Inspection Control 20-7
  - Verifying and Monitoring DNS Inspection 20-10

- FTP Inspection 20-11
  - FTP Inspection Overview 20-11
Contents

Using the **strict** Option 20-11
Configuring an FTP Inspection Policy Map for Additional Inspection Control 20-12
Verifying and Monitoring FTP Inspection 20-16

HTTP Inspection 20-16
ICMP Inspection 20-16
ICMP Error Inspection 20-16
Instant Messaging Inspection 20-17
  IM Inspection Overview 20-17
  Configuring an Instant Messaging Inspection Policy Map for Additional Inspection Control 20-17
IP Options Inspection 20-20
  IP Options Inspection Overview 20-20
  Configuring an IP Options Inspection Policy Map for Additional Inspection Control 20-21
IPsec Pass Through Inspection 20-22
  IPsec Pass Through Inspection Overview 20-22
  Example for Defining an IPsec Pass Through Parameter Map 20-23
NetBIOS Inspection 20-23
  NetBIOS Inspection Overview 20-23
  Configuring a NetBIOS Inspection Policy Map for Additional Inspection Control 20-23
PPTP Inspection 20-25
SMTP and Extended SMTP Inspection 20-25
  SMTP and ESMTP Inspection Overview 20-25
  Configuring an ESMTP Inspection Policy Map for Additional Inspection Control 20-26
TFTP Inspection 20-28

**CHAPTER 21**

**Configuring Inspection for Voice and Video Protocols** 21-1

CTIQBE Inspection 21-1
  CTIQBE Inspection Overview 21-1
  Limitations and Restrictions 21-2
  Verifying and Monitoring CTIQBE Inspection 21-2

H.323 Inspection 21-3
  H.323 Inspection Overview 21-4
  How H.323 Works 21-4
  H.239 Support in H.245 Messages 21-5
  Limitations and Restrictions 21-5
  Configuring an H.323 Inspection Policy Map for Additional Inspection Control 21-6
  Configuring H.323 and H.225 Timeout Values 21-9
  Verifying and Monitoring H.323 Inspection 21-9
    Monitoring H.225 Sessions 21-9
Monitoring H.245 Sessions  21-10
Monitoring H.323 RAS Sessions  21-10

MGCP Inspection  21-11
MGCP Inspection Overview  21-11
Configuring an MGCP Inspection Policy Map for Additional Inspection Control  21-13
Configuring MGCP Timeout Values  21-14
Verifying and Monitoring MGCP Inspection  21-14

RTSP Inspection  21-15
RTSP Inspection Overview  21-15
Using RealPlayer  21-16
Restrictions and Limitations  21-16
Configuring an RTSP Inspection Policy Map for Additional Inspection Control  21-16

SIP Inspection  21-19
SIP Inspection Overview  21-19
SIP Instant Messaging  21-19
Configuring a SIP Inspection Policy Map for Additional Inspection Control  21-20
Configuring SIP Timeout Values  21-24
Verifying and Monitoring SIP Inspection  21-24

Skinny (SCCP) Inspection  21-25
SCCP Inspection Overview  21-25
Supporting Cisco IP Phones  21-25
Restrictions and Limitations  21-26
Configuring a Skinny (SCCP) Inspection Policy Map for Additional Inspection Control  21-26
Verifying and Monitoring SCCP Inspection  21-28

CHAPTER 22
Configuring Inspection of Database and Directory Protocols  22-1

ILS Inspection  22-1
SQL*Net Inspection  22-2
Sun RPC Inspection  22-3
Sun RPC Inspection Overview  22-3
Managing Sun RPC Services  22-4
Verifying and Monitoring Sun RPC Inspection  22-4

CHAPTER 23
Configuring Inspection for Management Application Protocols  23-1

DCERPC Inspection  23-1
DCERPC Overview  23-1
Configuring a DCERPC Inspection Policy Map for Additional Inspection Control  23-2
RADIUS Accounting Inspection  23-3
PART 9  Configuring VPN

CHAPTER 26  Configuring IPsec and ISAKMP  26-1

Information About Tunneling, IPsec, and ISAKMP  26-1
IPsec Overview  26-2
ISAKMP and IKE Overview  26-2

Configuring ISAKMP  26-3
Conf iguring IKEv1 and IKEv2 Policies  26-3
Enabling IKE on the Outside Interface  26-7
Disabling IKEv1 Aggressive Mode  26-7
Determining an ID Method for IKEv1 and IKEv2 ISAKMP Peers  26-7
Enabling IPsec over NAT-T  26-8
Using NAT-T  26-8
Waiting for Active Sessions to Terminate Before Rebooting  26-9
Alerting Peers Before Disconnecting  26-9

Using the Tunnel-group-map default-group Command  26-9

Configuring IPsec  26-9
Understanding IPsec Tunnels  26-10
Understanding IKEv1 Transform Sets and IKEv2 Proposals  26-10
Defining Crypto Maps  26-11
Applying Crypto Maps to Interfaces  26-18
Using Interface Access Lists  26-18
Changing IPsec SA Lifetimes  26-20
Creating a Basic IPsec Configuration  26-21
Using Dynamic Crypto Maps  26-23
Providing Site-to-Site Redundancy  26-25
Viewing an IPsec Configuration  26-25
Clearing Security Associations  26-26
Clearing Crypto Map Configurations  26-26

CHAPTER 27  Configuring Connection Profiles, Group Policies, and Users  27-1

Overview of Connection Profiles, Group Policies, and Users  27-1

Connection Profiles  27-2
General Connection Profile Connection Parameters  27-2
Configuring Connection Profiles  27-3
Maximum Connection Profiles  27-3
Cisco ASA 1000V CLI Configuration Guide for ASDM Mode

Contents

Chapter 27
Configuring IPsec Tunnel-Group General Attributes 27-3
Configuring LAN-to-LAN Connection Profiles 27-4
  Default LAN-to-LAN Connection Profile Configuration 27-4
  Specifying a Name and Type for a LAN-to-LAN Connection Profile 27-4
  Configuring LAN-to-LAN Connection Profile General Attributes 27-4
  Configuring LAN-to-LAN IPsec IKEv1 Attributes 27-5

Group Policies 27-7
  Default Group Policy 27-7
  Configuring Group Policies 27-9
    Configuring an External Group Policy 27-9
    Configuring an Internal Group Policy 27-10
    Configuring Group Policy Attributes 27-10
    Configuring Site-to-Site VPN-Specific Attributes 27-10
    Configuring Domain Attributes for Tunneling 27-13
    Configuring Backup Server Attributes 27-15

Chapter 28
Configuring LAN-to-LAN IPsec VPNs 28-1
  Summary of the Configuration 28-2
  Configuring Interfaces 28-2
  Configuring ISAKMP Policy and Enabling ISAKMP on the Outside Interface 28-3
    Configuring ISAKMP Policies for IKEv1 Connections 28-4
    Configuring ISAKMP Policies for IKEv2 Connections 28-5
  Creating an IKEv1 Transform Set 28-5
  Creating an IKEv2 Proposal 28-6
  Configuring an ACL 28-7
  Defining a Tunnel Group 28-8
  Creating a Crypto Map and Applying It To an Interface 28-9
    Applying Crypto Maps to Interfaces 28-10

Part 10
Configuring Logging, SNMP, and Smart Call Home

Chapter 29
Configuring Logging 29-1
  Information About Logging 29-1
    Analyzing Syslog Messages 29-2
    Syslog Message Format 29-2
    Severity Levels 29-3
    Message Classes and Range of Syslog IDs 29-3
    Filtering Syslog Messages 29-3

Cisco ASA 1000V CLI Configuration Guide for ASDM Mode
# Using Custom Message Lists
29-4

## Prerequisites for Logging
29-4

## Guidelines and Limitations
29-5

## Configuring Logging
29-5

### Enabling Logging
29-5

### Configuring an Output Destination
29-6
- Sending Syslog Messages to an External Syslog Server 29-7
- Sending Syslog Messages to the Internal Log Buffer 29-8
- Sending Syslog Messages to an E-mail Address 29-9
- Sending Syslog Messages to ASDM 29-10
- Sending Syslog Messages to the Console Port 29-10
- Sending Syslog Messages to an SNMP Server 29-11
- Sending Syslog Messages to a Telnet or SSH Session 29-11

### Creating a Custom Event List
29-12

### Generating Syslog Messages in EMBLEM Format to a Syslog Server
29-13

### Generating Syslog Messages in EMBLEM Format to Other Output Destinations
29-13

### Changing the Amount of Internal Flash Memory Available for Logs
29-14

### Configuring the Logging Queue
29-14

### Sending All Syslog Messages in a Class to a Specified Output Destination
29-15

### Enabling Secure Logging
29-15

### Including the Device ID in Non-EMBLEM Format Syslog Messages
29-16

### Including the Date and Time in Syslog Messages
29-16

### Disabling a Syslog Message
29-17

### Changing the Severity Level of a Syslog Message
29-17

### Limiting the Rate of Syslog Message Generation
29-17

## Monitoring the Logs
29-18

## Configuration Examples for Logging
29-18

## Feature History for Logging
29-19

---

# Configuring SNMP
30-1

## Information About SNMP
30-1

### Information About SNMP Terminology
30-2

### Information About MIBs and Traps
30-3

### SNMP Object Identifiers
30-3

### SNMP Physical Vendor Type Values
30-3

### Supported Tables and Objects in MIBs
30-4

### Supported Traps (Notifications)
30-4

### SNMP Version 3
30-6

### SNMP Version 3 Overview
30-6
Contents

Cisco ASA 1000V CLI Configuration Guide for ASDM Mode

Pinging ASA 1000V Interfaces 33-3
Passing Traffic Through the ASA 1000V 33-5
Disabling the Test Configuration 33-7
Determining Packet Routing with Traceroute 33-7
Tracing Packets with Packet Tracer 33-7
Handling TCP Packet Loss 33-8
Reloading the ASA 1000V 33-8
Performing Password Recovery 33-8
   Recovering Passwords or Images on the ASA 1000V 33-9
   Disabling Password Recovery 33-9
Erasing the Flash File System 33-9
Other Troubleshooting Tools 33-10
   Viewing Debugging Messages 33-10
   Capturing Packets 33-11
   Viewing the Crash Dump 33-11
   Coredump 33-11
   Monitoring Per-Process CPU Usage 33-11

Reference

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
Supported Character Sets     A-7

APPENDIX B

Addresses, Protocols, and Ports     B-1

IPv4 Addresses and Subnet Masks     B-1

Classes     B-1
Private Networks     B-2
Subnet Masks     B-2

Determining the Subnet Mask     B-3
Determining the Address to Use with the Subnet Mask     B-3

Protocols and Applications     B-5
TCP and UDP Ports     B-5
Local Ports and Protocols     B-8
ICMP Types     B-9

APPENDIX C

Configuring an External Server for Authorization and Authentication     C-1

Understanding Policy Enforcement of Permissions and Attributes     C-1

Configuring an External LDAP Server     C-2

Organizing the ASA 1000V for LDAP Operations     C-3
Searching the LDAP Hierarchy     C-3
Binding the ASA 1000V to the LDAP Server     C-4
Defining the ASA 1000V LDAP Configuration     C-5
Supported Cisco Attributes for LDAP Authorization     C-5
Cisco AV Pair Attribute Syntax     C-13
Cisco AV Pairs ACL Examples     C-14
Active Directory/LDAP VPN Remote Access Authorization Examples     C-16
User-Based Attributes Policy Enforcement     C-16
Placing LDAP Users in a Specific Group Policy     C-18
Enforcing Static IP Address Assignment for AnyConnect Tunnels     C-20
Enforcing Dial-in Allow or Deny Access     C-22
Enforcing Logon Hours and Time-of-Day Rules     C-25

Configuring an External RADIUS Server     C-27
Reviewing the RADIUS Configuration Procedure     C-27
ASA 1000V RADIUS Authorization Attributes     C-27
ASA 1000V IETF RADIUS Authorization Attributes     C-36

Configuring an External TACACS+ Server     C-37

GLOSSARY

INDEX
About This Guide

This preface introduces the Cisco ASA 1000V CLI Configuration Guide for ASDM Mode and includes the following sections:

- Document Objectives, page xxv
- Audience, page xxv
- Related Documentation, page xxv
- Conventions, page xxvi
- Obtaining Documentation and Submitting a Service Request, page xxvii

Document Objectives

The purpose of this guide is to help you configure the ASA 1000V 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 ASA 1000V 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 only to the ASA 1000V when you are using the ASDM mode.

Audience

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

- Manage network security
- Install and configure firewalls/ASA 1000Vs
- Configure IPsec site-to-site VPN tunnels

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.
P A R T  1

Getting Started with the ASA 1000V
Introduction to the Cisco ASA 1000V

The ASA 1000V is an edge firewall virtual appliance that runs on VMware vSphere Hypervisor software and the Cisco Nexus 1000V switch exclusively. It allows Virtual Machines (VMs) in a virtual data center to access the Internet securely (including inter-tenant communications), functions as the default gateway for the VMs, and protects against network-based attacks.

This guide describes how to configure the ASA 1000V using ASDM mode (in this mode, you can use the ASA CLI); if you are using VNMC mode, see the VNMC documentation.

This chapter includes the following sections:

- New Features in ASA 1000V Version 8.7(1), page 1-1
- Supported and Unsupported Features of the Cisco ASA 1000V, page 1-3
- VMware Feature Support for the ASA 1000V, page 1-5
- Task Flow for Configuring the ASA 1000V, page 1-5
- Cloning the ASA 1000V, page 1-6
- Licensing Enforcement for the ASA 1000V, page 1-7
- Configuration Examples for the ASA 1000V, page 1-8
- Firewall Functional Overview, page 1-8
- IPsec Site-to-Site VPN Functional Overview, page 1-11
- Additional References, page 1-11

New Features in ASA 1000V Version 8.7(1)

Table 1-1 lists the new features for ASA 1000V Version 8.7(1).

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Platform Features</td>
<td></td>
</tr>
<tr>
<td>Support for the ASA 1000V</td>
<td>We introduced support for the ASA 1000V with the Cisco Nexus 1000V switch.</td>
</tr>
</tbody>
</table>

Table 1-1 New Features for ASA 1000V Version 8.7(1)
### Table 1-1 New Features for ASA 1000V Version 8.7(1) (continued)

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloning the ASA 1000V</td>
<td>You can add one or multiple instances of the ASA 1000V to your deployment using the method of cloning VMs.</td>
</tr>
</tbody>
</table>

#### Management Features

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASDM mode</td>
<td>You can configure, manage, and monitor the ASA 1000V using the Adaptive Security Device Manager (ASDM), which is the single GUI-based device manager for the ASA 1000V.</td>
</tr>
<tr>
<td>VNMC mode</td>
<td>You can configure and manage the ASA 1000V using the Cisco Virtual Network Management Center (VNMC), which is a GUI-based multi-device manager for multiple tenants.</td>
</tr>
<tr>
<td>XML APIs</td>
<td>You can configure and manage the ASA 1000V using XML APIs, which are application programmatic interfaces provided through the Cisco VNMC. This feature is only available in VNMC mode.</td>
</tr>
</tbody>
</table>

#### Firewall Features

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco VNMC access and configuration</td>
<td>Cisco VNMC access and configuration are required to create security profiles. You can configure access to the Cisco VNMC through the Configuration &gt; Device Setup &gt; Interfaces pane. Enter the login username and password, hostname, and shared secret to access the Cisco VNMC. Then you can configure security profiles and security profile interfaces. In VNMC mode, use the CLI to configure security profiles.</td>
</tr>
<tr>
<td>Security profiles and security profile interfaces</td>
<td>Security profiles are interfaces that correspond to an edge security profile that has been configured in Cisco VNMC and assigned in the Cisco Nexus 1000V VSM. Policies for through-traffic are assigned to these interfaces and the outside interface. You can add security profiles through the Configuration &gt; Device Setup &gt; Interfaces pane. You create the security profile by adding its name and selecting the service interface. ASDM then generates the security profile through the Cisco VNMC, assigns the security profile ID, and automatically generates a unique interface name. The interface name is used in the security policy configuration. We introduced or modified the following commands: interface security-profile, security-profile, mtu, vpath path-mtu, clear interface security-profile, clear configure interface security-profile, show interface security-profile, show running-config interface security-profile, show interface ip brief, show running-config mtu, show vsn ip binding, show vsn security-profile.</td>
</tr>
<tr>
<td>Service interface</td>
<td>The service interface is the Ethernet interface associated with security profile interfaces. You can only configure one service interface, which must be the inside interface. We introduced the following command: service-interface security-profile all.</td>
</tr>
<tr>
<td>VNMC policy agent</td>
<td>The VNMC policy agent enables policy configuration through both the ASDM and VNMC modes. It includes a web server that receives XML-based requests from Cisco VNMC over HTTPS and converts it to the ASA 1000V configuration. We introduced the following commands: vnmc policy-agent, login, shared-secret, registration host, vnmc org, show vnmc policy-agent, show running-config vnmc policy-agent, clear configure vnmc policy-agent.</td>
</tr>
</tbody>
</table>
Supported and Unsupported Features of the Cisco ASA 1000V

The ASA 1000V supports a subset of features of the ASA. Table 1-2 lists the major supported features on the ASA 1000V.

Table 1-2  Major Supported Features on the ASA 1000V

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AAA for management access</td>
<td>See Chapter 17, “Configuring Management Access.”</td>
</tr>
<tr>
<td>DHCP server, DHCP client, and DHCP relay</td>
<td>See Chapter 6, “Configuring DHCP.”</td>
</tr>
<tr>
<td>DNS server for name resolution</td>
<td>See Chapter 5, “Configuring the Hostname, Domain Name, Passwords, and Other Basic Settings.”</td>
</tr>
<tr>
<td>Failover</td>
<td>Active/Standby only. See Chapter 3, “Configuring Active/Standby Failover.”</td>
</tr>
<tr>
<td>IPsec site-to-site VPN</td>
<td>Static tunnels only. See Chapter 28, “Configuring LAN-to-LAN IPsec VPNs.”</td>
</tr>
<tr>
<td>NTP and time zones</td>
<td>See Chapter 5, “Configuring the Hostname, Domain Name, Passwords, and Other Basic Settings.”</td>
</tr>
<tr>
<td>SNMP MIBs and traps</td>
<td>See Chapter 30, “Configuring SNMP.”</td>
</tr>
<tr>
<td>SSH and Telnet</td>
<td>See Chapter 17, “Configuring Management Access.”</td>
</tr>
<tr>
<td>Syslog messages (TCP and UDP)</td>
<td>See the syslog messages guide and Chapter 29, “Configuring Logging.”</td>
</tr>
<tr>
<td>TCP intercept</td>
<td>See Chapter 24, “Configuring Connection Settings.”</td>
</tr>
</tbody>
</table>

Table 1-3 lists the unsupported features on the ASA 1000V.

Note  The commands associated with an unsupported feature are not supported at the CLI.

Table 1-3  Unsupported Features on the ASA 1000V

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AAA for network access</td>
<td>Not supported.</td>
</tr>
<tr>
<td>Active/Active failover and subsecond failover</td>
<td>Not supported.</td>
</tr>
<tr>
<td>Authentication using certificates</td>
<td>Not supported.</td>
</tr>
</tbody>
</table>
Supported and Unsupported Features of the Cisco ASA 1000V

For more information about the ASA 1000V, see the following URL:

VMware Feature Support for the ASA 1000V

Table 1-4 lists the VMware feature support for the ASA 1000V.

**Table 1-4** VMware Feature Support for the ASA 1000V

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
<th>Support (Yes/No)</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cold clone</td>
<td>The VM is powered off before cloning.</td>
<td>Yes</td>
<td>—</td>
</tr>
<tr>
<td>DRS</td>
<td>Used for dynamic resource scheduling and distributed power management.</td>
<td>Yes</td>
<td>—</td>
</tr>
<tr>
<td>Hot clone</td>
<td>The VM is running during cloning.</td>
<td>No</td>
<td>—</td>
</tr>
<tr>
<td>Snapshot</td>
<td>Freezes the VM for a few seconds. You may lose traffic. Failover may occur.</td>
<td>See comment.</td>
<td>Use with care.</td>
</tr>
<tr>
<td>VM migration</td>
<td>Used for VM migration.</td>
<td>Yes</td>
<td>—</td>
</tr>
<tr>
<td>vMotion</td>
<td>Used for live migration of VMs.</td>
<td>Yes</td>
<td>—</td>
</tr>
<tr>
<td>VMware FT</td>
<td>Used for HA for VMs.</td>
<td>No</td>
<td>Use ASA 1000V failover for ASA 1000V VM failures.</td>
</tr>
<tr>
<td>VMware HA</td>
<td>Used for ESX and server failures.</td>
<td>Yes</td>
<td>Use ASA 1000V failover for ASA 1000V VM failures.</td>
</tr>
<tr>
<td>VMware HA with VM heartbeats</td>
<td>Used for VM failures.</td>
<td>No</td>
<td>Use ASA 1000V failover for ASA 1000V VM failures.</td>
</tr>
</tbody>
</table>

Task Flow for Configuring the ASA 1000V

Table 1-5 lists the major tasks for configuring the ASA 1000V. This guide also includes other optional features not listed in this table.

**Table 1-5** Task Flow for Configuring the ASA 1000V

<table>
<thead>
<tr>
<th>Task</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Configuring the VSN.</td>
<td>See the <em>Cisco Nexus 1000V Interface Configuration Guide</em>.</td>
</tr>
<tr>
<td>2. Deploying the ASA 1000V.</td>
<td>See the <em>Cisco ASA 1000V Getting Started Guide</em>.</td>
</tr>
<tr>
<td>5. Configuring interfaces.</td>
<td>See Chapter 4, “Configuring Interfaces.”</td>
</tr>
<tr>
<td>7. Configuring service policies.</td>
<td>See Chapter 14, “Configuring a Service Policy Using the Modular Policy Framework.”</td>
</tr>
</tbody>
</table>
Cloning the ASA 1000V

You can deploy multiple instances of the ASA 1000V. You can add one or multiple instances of the ASA 1000V to your deployment by using either of the following methods:

- **Factory-shipped OVA template:** This method enables you to clone multiple instances of the ASA 1000V with different sets of OVF deployment parameters. The ASA 1000Vs have blank configurations, and you obtain the required configuration settings for each instance from the OVF deployment parameters.

- **Configured OVA template:** This method enables you to clone a previously configured OVA template. The ASA 1000Vs already have configurations, and you simply reapply the current OVF parameters. The result is that you can clone multiple instances of ready-to-use ASA 1000V instances with the required configurations.

- **VMware vSphere API:** This method enables you to use application programmatic interfaces to clone an existing ASA 1000V and change the current configuration parameters for each new ASA 1000V instance.

Requirements for Cloning

This section includes the following topics:

- Preparing the Master ASA 1000V, page 1-6
- Cloning the Master, page 1-6

Preparing the Master ASA 1000V

To prepare the master ASA 1000V, perform the following steps:

1. Deploy the ASA 1000V, set up management routes, SSH access, the username and passwords, and ASDM access.
2. Register with the VNMC using the steps listed in the Cisco ASA 1000V Getting Started Guide.
3. Save the running configuration to the startup configuration using the `write memory` command.
4. It is critical that you power off the ASA 1000V.

   **Note** If you clone a powered-on ASA 1000V, there is a slight chance that the original ASA 1000V may crash right after the cloning is completed.

Cloning the Master

To clone the master ASA 1000V, perform the following steps:

1. Export the master ASA 1000V to an OVA template using the vSphere Client GUI or any other equivalent method. For example, you can use ovftool from VMware as follows:
   
   ```bash
   ovftool vi://sample-vc50.example.com/performance-testbed/vm/ASA_1000V asa_master.ova
   ```

   **Note** Using vSphere 4.x to create an OVA template removes the vApp properties set in the master. This means that when the clone is deployed in Step 2, all vApp properties need to be specified.
2. Create a cloned ASA 1000V instance using the VSphere Client GUI or any other equivalent method. For example, you can use ovftool from VMware as follows:

```
    ovftool --acceptAllEulas --datastore="datastore1 (1)" --net:VM Network=525-net-vm
    "--net:425-net-vm=60-net" "--net:60-net=425-net-vm" --prop:ManagementIPv4=2.2.2.5
    --prop:ManagementIPv4Gateway=2.2.2.2 --prop:ManagementIPv4Subnet=255.255.0.0 --name="ASA
    Clone 2 " asa_master.ova
    vi://sample-vc50.example.com/performance-testbed/host/sample-esxhost.example.com/
```

It is important to change the Management interface IP address in the clone so that it becomes a separate instance. Otherwise, an IP address conflict will occur. All other vApp properties could be changed if different parameters need to be set. Note that if the OVA template is created using VSphere 4.x, then all vApp properties need to be specified in the clone, because these properties are not stored in the template.

Note
The ovftool does not export default values if you are using an ESXi 4.1 deployment; however, an ESXi 5.0 deployment does support the export of default values.

The following is an example using ovftool from VMware to set all of the properties:

```
    ovftool --acceptAllEulas --datastore="datastore1 (1)" --net:VM Network=525-net-vm
    "--net:425-net-vm=60-net" "--net:60-net=425-net-vm" --prop:ManagementIPv4=2.2.2.5
    --prop:ManagementIPv4Gateway=2.2.2.2 --prop:ManagementIPv4Subnet=255.255.0.0
    --prop:ASDMIPv4=0.0.0.0 --prop:HAActiveIPv4=0.0.0.0 --prop:HASubnetIPv4=0.0.0.0
    --prop:HAStandbyIPv4=0.0.0.0 --prop:ManagementStandbyIPv4=0.0.0.0
    --prop:VNMCIPv4=0.0.0.0 --name="ASA Clone 2 " asa_master.ova
    vi://sample-vc50.example.com/performance-testbed/host/sample-esxhost.example.com/
```

Fields marked with 0.0.0.0 values should be set appropriately if used; otherwise, they can be set as 0.0.0.0.

3. Power on the clone.

Note
Cloned ASA 1000Vs regenerate default RSA keypairs with the same settings and delete all other keys.

### Licensing Enforcement for the ASA 1000V

The Nexus 1000V Virtual Service Module (VSM) requires a license that controls the number of CPU sockets on each Virtual Ethernet Module (VEM) used for the ASA 1000V. If the VSM does not have enough licenses, and you deploy an ASA 1000V without license support, then traffic is not allowed to pass through the ASA 1000V. This means the following:

- For traffic passing from inside to outside, traffic never reaches the ASA 1000V. See syslog 4450002 for more information.
- For traffic passing from outside to inside, the ASA 1000V allows the initial packet to pass through, but the vPath module on the Nexus 1000V rejects the packet, and the ASA 1000V deletes the flow. See syslog 4450002 for more information.
Configuration Examples for the ASA 1000V

Specific feature configuration examples are provided in the Cisco ASA 1000V Getting Started Guide.

Monitoring the ASA 1000V

You can monitor the ASA 1000V in both ASDM mode and VNMC mode (by accessing the monitoring panes in ASDM in the read-only view), as well as through SSH, Telnet, or the console. The ASA 1000V commands that we have introduced or modified for this platform are included in the feature history table at the end of the chapter for each individual feature in this guide.

Firewall Functional Overview

Firewalls protect inside networks from unauthorized access by users on an outside network. Firewalls enable you to control when inside users access outside networks (for example, access to the Internet), by allowing only certain addresses out.

When describing networks connected to a firewall, the outside network is in front of the firewall, and the inside network is protected and behind the firewall. The ASA 1000V allows two data interfaces: one inside and one outside, plus a management interface and one interface for failover. The ASA 1000V allows the creation of multiple security profile interfaces to provide different security policies for the virtual machines (VMs) on the inside interface when they access the outside. (Traffic between VMs is controlled through the use of the Cisco VSG).

The maximum number of concurrent firewall connections allowed on the ASA 1000V is 200,000.

This section includes the following topics:

- Security Policy Overview, page 1-8
- Failover Overview, page 1-9
- Firewall Mode Overview, page 1-10
- Stateful Inspection Overview, page 1-10

Security Policy Overview

A security policy determines which traffic is allowed to pass through the firewall to access another network. By default, the ASA 1000V allows traffic to flow freely from a security profile interface on 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 (Rules), page 1-9
- Applying NAT, page 1-9
- Protection from IP Fragments, page 1-9
- Applying Application Inspection, page 1-9
- Applying Connection Limits and TCP Normalization, page 1-9
Permitting or Denying Traffic with Access Lists (Rules)

You can apply an access rule to limit traffic from the inside security profile to the outside or to allow traffic from the outside to the inside security profile.

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.

Protection from IP Fragments

The ASA 1000V 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 ASA 1000V. Fragments that fail the security check are dropped and logged. Virtual reassembly cannot be disabled.

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.

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 ASA 1000V uses the embryonic limit to trigger TCP Intercept, which protects inside systems from a DoS attack that is perpetrated by flooding an interface with TCP SYN packets. An embryonic connection is a connection request that has not finished the necessary handshake between the source and the destination.

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

Failover Overview

Configuring high availability requires two identical ASA 1000Vs connected to each other through a dedicated Stateful Failover link. The two ASA 1000Vs in a failover pair constantly communicate over a failover link to determine the operating status of each one. The health of the active interfaces is monitored to determine if specific failover conditions are met. If those conditions are met, failover occurs.

You can use the GigabitEthernet 0/2 interface on the ASA 1000V as the failover link. 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).
The ASA 1000V only supports Active/Standby failover, in which one ASA 1000V passes traffic while the other ASA 1000V waits in a standby state.

Firewall Mode Overview

The ASA 1000V runs only in routed firewall mode and is considered to be a router hop in the network.

Stateful Inspection Overview

All traffic that goes through the ASA 1000V 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 with 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 24-3.

A stateful firewall like the ASA 1000V, however, takes into consideration the state of a packet:

- Is this a new connection?
  
  If it is a new connection, the ASA 1000V 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
  - Performing route lookups
  - Allocating NAT translations (xlates)
  - Establishing sessions in the fast path

  Some packets that require Layer 7 inspection (the packet payload must be inspected or altered) are passed on to the control plane path. Layer 7 inspection engines are required for protocols that have two or more channels: a data channel, which uses well-known port numbers, and a control channel, which uses different port numbers for each session. These protocols include FTP, H.323, and SNMP.

- Is this an established connection?
  
  If the connection is already established, the ASA 1000V does not need to recheck packets; most matching packets can go through the fast path in both directions. The fast path is responsible for the following tasks:
  
  - Verifying the IP checksum
  - Performing session lookups
  - Checking TCP sequence numbers
  - Allocating NAT translations based on existing sessions
  - Adjusting Layer 3 and Layer 4 headers

  For UDP or other connectionless protocols, the ASA 1000V creates connection state information so that it can also use the fast path.
Data packets for protocols that require Layer 7 inspection can also go through the fast path. Some established session packets must continue to go through the session management path or the control plane path. Packets that go through the session management path include HTTP packets that require inspection or content filtering. Packets that go through the control plane path include the control packets for protocols that require Layer 7 inspection.

**IPsec Site-to-Site VPN Functional Overview**

A site-to-site (LAN-to-LAN) VPN connects networks in different geographic locations. The ASA 1000V supports IPsec site-to-site connections (called tunnels) to Cisco or third-party peers when the two peers have IPv4 inside and outside interfaces. IPsec tunnel mode is useful for protecting traffic between different networks when traffic must pass through an intermediate, untrusted network.

The supported protocols for IPsec site-to-site tunnels are IKEv1 and IKEv2 using a preshared key only. A preshared key or shared secret is the string of text that a VPN expects to receive before any other credentials (such as a username and password). Split tunnels are also supported and allow the network administrator to determine which traffic the client passes through the VPN tunnel and which traffic goes straight to the Internet (non-tunneled).

The supported tunnel modes are Encapsulating Security Payload (ESP) alone, and ESP and Authentication Header (AH) combined. AH tunnel mode encapsulates an IP packet with an AH and IP header and signs the entire packet for integrity and authentication. ESP tunnel mode encapsulates an IP packet with both an ESP and IP header and an ESP authentication trailer. ESP and AH can be combined when tunneling, which provides both security for the tunneled IP packet and integrity and authentication for the entire packet.

In addition, NAT traversal is supported and is used by IPsec site-to-site VPN clients to allow ESP packets to traverse NAT.

**Additional References**

For more information about the individual components that comprise the ASA 1000V, see the following documentation:

- ASA 1000V
- ASDM
- Cisco Nexus 1000V
- Cisco VNMC and Cisco VSG
- VMware
Getting Started

This chapter describes how to get started with your ASA 1000V. This guide covers ASDM mode configuration, where you can use ASDM and the ASA 1000V CLI for your configuration. If you are using VNMC mode, see the VNMC documentation to complete your configuration.

As part of your deployment, you pre-configure the management interface for ASDM access. This chapter includes information to help you get started with the CLI.

This chapter includes the following sections:

- Deploying and Powering Up the ASA 1000V, page 2-1
- Accessing the ASA 1000V Command-Line Interface, page 2-2
- Starting ASDM, page 2-3
- ASA 1000V File Storage, page 2-7
- Working with the Configuration, page 2-7
- Applying Configuration Changes to Connections, page 2-10

Deploying and Powering Up the ASA 1000V

For detailed instructions on deploying and powering up the ASA 1000V, see the Getting Started Guide.

Default Configuration after Deployment

When you deploy the ASA 1000V, you can pre-set many parameters that let you connect to the Management 0/0 interface using ASDM. A typical configuration includes the following settings:

- Management 0/0 interface:
  - Named “management”
  - IP address or DHCP
  - Security level 0
  - Management-only
- Static route from the management interface to the management host IP address through the default gateway
- Static route from the management interface to the VNMC IP address through the default gateway
Accessing the ASA 1000V Command-Line Interface

For initial configuration or troubleshooting, access the CLI from the virtual console provided through the VMware vSphere Client. Later, you can configure CLI remote access using Telnet or SSH according to Chapter 17, “Configuring Management Access.”

Detailed Steps

Step 1  In the VMware vSphere Client, choose Home > Inventory > Hosts and Clusters, and then choose the ASA 1000V instance that you deployed and powered up.

Step 2  In the right pane, click the Console tab.

Step 3  You see the following prompt:

hostname>

This prompt indicates that you are in user EXEC mode. Only basic commands are available from user EXEC mode.

Step 4  To access privileged EXEC mode, enter the following command:

hostname> enable

The following prompt appears:

Password:

- ASDM server enabled
- ASDM access for the management host IP address

See the following configuration:

```
interface Management0/0
  nameif management
  security-level 0
  ip address ip_address
  management-only
  route management management_host_IP mask gateway_ip
  route management VNMC_IP mask gateway_ip
  http server enable
  http management_host_IP mask management
```
Starting ASDM

You can start ASDM using two methods:

- **ASDM-IDM Launcher (Windows only)**—The Launcher is an application downloaded from the ASA 1000V using a web browser that you can use to connect to any ASA 1000V IP address. You do not need to re-download the launcher if you want to connect to other ASA 1000Vs. The Launcher also lets you run a virtual ASDM in Demo mode using files downloaded locally.

- **Java Web Start**—For each ASA 1000V 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 ASA 1000V IP address.

**Note**

Within ASDM, you can choose a different ASA 1000V IP address to manage; the difference between the Launcher and Java Web Start application functionality rests primarily in how you initially connect to the ASA 1000V 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](#)
- [Starting ASDM from the ASDM-IDM Launcher](#)
- [Starting ASDM from the Java Web Start Application](#)
- [Using ASDM in Demo Mode](#)
ASDM allows multiple PCs or workstations to each have one browser session open with the same ASA 1000V software. A single ASA 1000V can support up to five concurrent ASDM sessions. Only one session per browser per PC or workstation is supported for a specified ASA 1000V.

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 ASA 1000V network, enter the following URL:

```
https://interface_ip_address/admin
```

Where `interface_ip_address` is the management IP address of the ASA 1000V.

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 2-5 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 2-5 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 2-4.

**Detailed Steps**

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Start the ASDM-IDM Launcher application.</td>
</tr>
<tr>
<td>2</td>
<td>Enter or choose the ASA 1000V 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.</td>
</tr>
<tr>
<td>3</td>
<td>Enter your username and your password, and then click <strong>OK</strong>. 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 <strong>enable</strong> password, which is blank by default. With HTTPS authentication enabled, enter your username and associated password.</td>
</tr>
<tr>
<td></td>
<td>If there is a new version of ASDM on the ASA 1000V, the ASDM Launcher automatically downloads the new version and requests that you update the current version before starting ASDM.</td>
</tr>
<tr>
<td></td>
<td>The main ASDM window appears.</td>
</tr>
</tbody>
</table>

Starting ASDM from the Java Web Start Application

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

**Prerequisites**

Download the Java Web Start application according to the “Connecting to ASDM for the First Time” section on page 2-4.

**Detailed Steps**

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Start the Java Web Start application.</td>
</tr>
<tr>
<td>2</td>
<td>Accept any certificates according to the dialog boxes that appear. The Cisco ASDM-IDM Launcher appears.</td>
</tr>
<tr>
<td>3</td>
<td>Enter the username and password, and click <strong>OK</strong>. 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 <strong>enable</strong> password, which is blank by default. With HTTPS authentication enabled, enter your username and associated password.</td>
</tr>
<tr>
<td></td>
<td>The main ASDM window appears.</td>
</tr>
</tbody>
</table>
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 ASA 1000V features using the ASDM interface.
- 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 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:
    - 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:
  - 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.

ASA 1000V File Storage

During OVF template file deployment, 2 GB of storage are allotted to maintain system, configuration, and image files on the host server. These files appear in disk0 on the ASA 1000V.

Working with the Configuration

This section describes how to work with the configuration. The ASA 1000V 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 32, “Managing Software and Configurations.”)

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.

This section includes the following topics:

- Saving Configuration Changes, page 2-7
- Copying the Startup Configuration to the Running Configuration, page 2-8
- Viewing the Configuration, page 2-8
- Clearing and Removing Configuration Settings, page 2-9
- Creating Text Configuration Files Offline, page 2-9

Saving Configuration Changes

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>

Example:

hostname# write memory

Note  The copy running-config startup-config command is equivalent to the write memory command.
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><code>copy startup-config running-config</code></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 ASA 1000V, then the effect of the merge depends on the command. You might get errors, or you might have unexpected results.</td>
</tr>
<tr>
<td><code>reload</code></td>
<td>Reloads the ASA 1000V, which loads the startup configuration and discards the running configuration.</td>
</tr>
<tr>
<td><code>clear configure all</code></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><code>show running-config</code></td>
<td>Views the running configuration.</td>
</tr>
<tr>
<td><code>show running-config command</code></td>
<td>Views the running configuration of a specific command.</td>
</tr>
<tr>
<td><code>show startup-config</code></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><code>clear configure configurationcommand [level2configurationcommand]</code></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 <code>level2configurationcommand</code>. For example, to clear the configuration for all <code>aaa</code> commands, enter the following command: <code>hostname(config)# clear configure aaa</code> To clear the configuration for only <code>aaa authentication</code> commands, enter the following command: <code>hostname(config)# clear configure aaa authentication</code></td>
</tr>
<tr>
<td><code>no configurationcommand [level2configurationcommand] qualifier</code></td>
<td>Disables the specific parameters or options of a command. In this case, you use the <code>no</code> command to remove the specific configuration identified by <code>qualifier</code>. For example, to remove a specific <code>nat</code> command, enter enough of the command to identify it uniquely as follows: <code>hostname(config)# no nat (inside) 1</code></td>
</tr>
<tr>
<td><code>write erase</code></td>
<td>Erases the startup configuration.</td>
</tr>
<tr>
<td><code>clear configure all</code></td>
<td>Erases the running configuration.</td>
</tr>
</tbody>
</table>

Creating Text Configuration Files Offline

This guide describes how to use the CLI to configure the ASA 1000V; 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 ASA 1000V internal flash memory. See Chapter 32, “Managing Software and Configurations,” for information on downloading the configuration file to the ASA 1000V.

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)# hostname a`

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

`hostname a`
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 show local-host all 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 all keyword. To clear connections to and from a particular IP address, use the ip_address argument.</td>
</tr>
<tr>
<td>clear conn [all] [protocol {tcp</td>
<td>udp}</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>

Example:

For additional information about formatting the file, see Appendix A, “Using the Command-Line Interface.”
PART 2

Configuring Basic Settings
Configuring Active/Standby Failover

This chapter describes how to configure Active/Standby failover and includes the following sections:

- Introduction to Failover and High Availability, page 3-1
- Prerequisites for Active/Standby Failover, page 3-14
- Guidelines and Limitations, page 3-15
- Configuring Active/Standby Failover, page 3-15
- Controlling Failover, page 3-22
- Monitoring Active/Standby Failover, page 3-24
- Feature History for Active/Standby Failover, page 3-24

Introduction to Failover and High Availability

Configuring high availability requires two identical ASA 1000Vs connected to each other through a dedicated failover link. The health of the active interfaces and ASA 1000Vs is monitored to determine if specific failover conditions are met. If those conditions are met, failover occurs.

The ASA 1000V supports Active/Standby failover. With Active/Standby failover, only one ASA 1000V passes traffic while the other ASA 1000V waits in a standby state. The interface typically assigned for active/standby failover pairs on the ASA 1000V is GigabitEthernet0/2.

- Failover System Requirements, page 3-2
- Failover and Stateful Failover Links, page 3-6
- Stateless (Regular) and Stateful Failover, page 3-9
- Auto Update Server Support in Failover Configurations, page 3-10
- Failover Health Monitoring, page 3-12
- Failover Messages, page 3-14
Failover System Requirements

Each ASA 1000V must be installed on separate Nexus 1000Vs.

The two ASA 1000Vs in a failover configuration 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 and have failover remain active. We recommend upgrading both ASA 1000Vs to the same version to ensure long-term compatibility.

See the “Performing Zero Downtime Upgrades for Failover Pairs” section on page 32-5 for more information about upgrading the software on a failover pair.

Information About Active/Standby Failover

This section describes Active/Standby failover and includes the following topics:

- Active/Standby Failover Overview, page 3-2
- Primary/Secondary Status and Active/Standby Status, page 3-2
- Device Initialization and Configuration Synchronization, page 3-3
- Command Replication, page 3-3
- Failover Triggers, page 3-4
- Failover Actions, page 3-5

Active/Standby Failover Overview

Active/Standby failover enables you to use a standby ASA 1000V to take over the functionality of a failed ASA 1000V. When the active ASA 1000V fails, it changes to the standby state while the standby ASA 1000V changes to the active state. The ASA 1000V that becomes active assumes the IP addresses and MAC addresses of the failed ASA 1000V and begins passing traffic. The ASA 1000V 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.

Primary/Secondary Status and Active/Standby Status

The main differences between the two ASA 1000Vs in a failover pair are related to which ASA 1000V is active and which ASA 1000V is standby, namely which IP addresses to use and which ASA 1000V actively passes traffic.

However, a few differences exist between the ASA 1000Vs based on which one is primary (as specified in the configuration) and which one is secondary:

- The primary ASA 1000V always becomes the active ASA 1000V if both ASA 1000Vs start up at the same time (and are of equal operational health).
- The primary ASA 1000V MAC addresses are always coupled with the active IP addresses. The exception to this rule occurs when the secondary ASA 1000V is active and cannot obtain the primary ASA 1000V MAC addresses over the failover link. In this case, the secondary ASA 1000V 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 ASA 1000V to the standby ASA 1000V. When the standby ASA 1000V completes its initial startup, it clears its running configuration (except for the failover commands needed to communicate with the active ASA 1000V, and the active ASA 1000V sends its entire configuration to the standby ASA 1000V.

The active ASA 1000V is determined by the following:

- If an ASA 1000V boots and detects a peer already running as active, it becomes the standby ASA 1000V.
- If an ASA 1000V boots and does not detect a peer, it becomes the active ASA 1000V.
- If both ASA 1000Vs boot simultaneously, then the primary ASA 1000V becomes the active ASA 1000V, and the secondary ASA 1000V becomes the standby ASA 1000V.

**Note**

If the secondary ASA 1000V boots without detecting the primary ASA 1000V, it becomes the active ASA 1000V. It uses its own MAC addresses for the active IP addresses. However, when the primary ASA 1000V becomes available, the secondary ASA 1000V changes the MAC addresses to those of the primary ASA 1000V, 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 3-21 for more information.

When the replication starts, the console on the active ASA 1000V displays the message “Beginning configuration replication: Sending to mate,” and when it is complete, the ASA 1000V displays the message “End Configuration Replication to mate.” During replication, commands entered on the active ASA 1000V may not replicate properly to the standby ASA 1000V, and commands entered on the standby ASA 1000V may be overwritten by the configuration being replicated from the active ASA 1000V. Avoid entering commands on either ASA 1000V 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 standby ASA 1000V, the configuration exists only in running memory. A failover pair of ASA 1000Vs must be deployed in the same mode (both VNMC or both ASDM) so that security policy configuration stays consistent across the failover pair. If the modes are mixed, the following error message appears:

> Mate's device manager mode (ASDM|VNMC) is not compatible with my mode (ASDM|VNMC). Failover will be disabled.

To save the configuration to flash memory after synchronization, enter the `write memory` command on the active ASA 1000V. The command is replicated to the standby ASA 1000V, which proceeds to write its configuration to flash memory.

Command Replication

Command replication always flows from the active ASA 1000V to the standby ASA 1000V. As commands are entered on the active ASA 1000V, they are sent across the failover link to the standby ASA 1000V. You do not have to save the active configuration to flash memory to replicate the commands.

The following commands that are replicated to the standby ASA 1000V:

- All configuration commands except for `mode`, `firewall`, and `failover lan unit`
- `copy running-config startup-config`
### Introduction to Failover and High Availability

- delete
- mkdir
- rename
- rmdir
- write memory

The following commands that are *not* replicated to the standby ASA 1000V:

- All forms of the `copy` command except for `copy running-config startup-config`
- All forms of the `write` command except for `write memory`
- debug
- failover lan unit
- firewall
- show
- `terminal pager` and `pager`

**Note** Changes made on the standby ASA 1000V are not replicated to the active ASA 1000V. If you enter a command on the standby ASA 1000V, the following message appears: ****** 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 ASA 1000V, the standby ASA 1000V clears its running configuration (except for the failover commands used to communicate with the active ASA 1000V), and the active ASA 1000V sends its entire configuration to the standby ASA 1000V.

Replicated commands are stored in the running configuration.

**Note** Standby Failover does not replicate the following files and configuration components:

- ASA 1000V images
- ASDM images

To save the replicated commands to the flash memory on the standby ASA 1000V, enter the `copy running-config startup-config` command on the active ASA 1000V. The command is replicated to the standby ASA 1000V, which proceeds to write its configuration to flash memory.

### Failover Triggers

The ASA 1000V can fail if one of the following events occurs:

- A hardware failure or a power failure occurs.
- A software failure occurs.
- Too many monitored interfaces fail.
- You force a failover. (See the “Forcing Failover” section on page 3-23.)
### Failover Actions

Table 3-1 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 ASA 1000V, the action taken by the standby ASA 1000V, and any special notes about the failover condition and actions.

**Table 3-1 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 ASA 1000V failed</td>
<td>Failover</td>
<td>n/a</td>
<td>Become active</td>
<td>No hello messages are received on any monitored interface or the failover link.</td>
</tr>
<tr>
<td>Formerly active ASA 1000V recovers</td>
<td>No failover</td>
<td>Become standby</td>
<td>No action</td>
<td>None.</td>
</tr>
<tr>
<td>Standby ASA 1000V failed</td>
<td>No failover</td>
<td>Mark standby as failed</td>
<td>n/a</td>
<td>When the standby ASA 1000V is marked as failed, then the active ASA 1000V 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 ASA 1000V cannot fail over to the standby ASA 1000V 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 ASA 1000Vs 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 ASA 1000V</td>
<td>Failover</td>
<td>Mark active as failed</td>
<td>Become active</td>
<td>None.</td>
</tr>
<tr>
<td>above threshold</td>
<td></td>
<td></td>
<td></td>
<td>When the standby ASA 1000V is marked as failed, then the active ASA 1000V does not attempt to fail over even if the interface failure threshold is surpassed.</td>
</tr>
<tr>
<td>Interface failure on standby ASA 1000V</td>
<td>No failover</td>
<td>No action</td>
<td>Mark standby as failed</td>
<td></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 an ASA 1000V and control which interfaces affect your failover.
- Interface health monitoring—Enables the ASA 1000V 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 ASA 1000V uses the correct MAC addresses when it is the active ASA 1000V, even if it comes online before the primary ASA 1000V.

Failover and Stateful Failover Links

This section describes the failover and the Stateful Failover links, which are dedicated connections between the two ASA 1000Vs in a failover configuration. This section includes the following topics:

- Failover Link, page 3-6
- Stateful Failover Link, page 3-7
- Avoiding Interrupted Failover Links, page 3-7

Failover Link

The two ASA 1000Vs in a failover pair constantly communicate over a failover link to determine the operating status of each ASA 1000V. The following information is communicated over the failover link:

- The ASA 1000V state (active or standby)
- Hello messages (keepalives)
- 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 the GigabitEthernet 0/2 interface on the device as the failover link. 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 using a switch, with no other device on the same network segment (broadcast domain or VLAN) as the failover interfaces of the ASA 1000V.
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. This is defined as the GigabitEthernet0/2 interface, with the name **FOlink**. This interface does not receive packets with vPath encapsulation.
  
  If you are managing policies through the ASDM, you can change interface roles after initial deployment.

- 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 by using a switch, with no other device on the same network segment (broadcast domain or VLAN) as the failover interfaces of the ASA 1000V.

**Note**

Enable the PortFast option on Cisco switch ports that connect directly to the ASA 1000V.

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**

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.

Avoiding Interrupted Failover Links

Because the ASA 1000V uses failover interfaces to transport messages between primary and secondary ASA 1000Vs, 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 ASA 1000V failover operation is affected until the health of the failover interface is restored.

In the event that all communication is cut off between the ASA 1000Vs in a failover pair, both ASA 1000Vs go into the active state, which is expected behavior. When communication is restored and the two active ASA 1000Vs resume communication through the failover link or through any monitored interface, the primary ASA 1000V remains active, and the secondary ASA 1000V immediately returns to the standby state. This relationship is established regardless of the health of the primary ASA 1000V.
Because of this behavior, stateful flows that were passed properly by the secondary active ASA 1000V 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 ASA 1000V takes a sample of the interface health, exchanges this information with its peer through the data interface, and performs a switchover if the active ASA 1000V 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 ASA 1000V 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 ASA 1000Vs, then when a switch or inter-switch-link is down, both ASA 1000Vs become active. Therefore, the following two connection methods shown in Figure 3-1 and Figure 3-2 are NOT recommended.

**Scenario 2—Recommended**
To make the ASA 1000V 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 to connect two ASA 1000V failover interfaces, as shown in Figure 3-3.
Scenario 3—Recommended
If the ASA 1000V 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 the network, as shown in Figure 3-4.

Stateless (Regular) and Stateful Failover
The ASA 1000V supports two types of failover, regular and stateful. This section includes the following topics:
- Stateless (Regular) Failover, page 3-9
- Stateful Failover, page 3-9

Stateless (Regular) Failover
When a failover occurs, all active connections are dropped. Clients need to reestablish connections when the new active ASA 1000V takes over.

Stateful Failover
When Stateful Failover is enabled, the active ASA 1000V continually passes per-connection state information to the standby ASA 1000V. After a failover occurs, the same connection information is available at the new active ASA 1000V. Supported end-user applications are not required to reconnect to keep the same communication session.

The following state information is passed to the standby ASA 1000V when Stateful Failover is enabled:
- NAT translation table
- TCP connection states
- UDP connection states
- The ARP table
- The HTTP connection states (if HTTP replication is enabled)
- The ISAKMP and IPsec SA table
- SIP signalling sessions
The following state information is not passed to the standby ASA 1000V when Stateful Failover is enabled:

- The HTTP connection table (unless HTTP replication is enabled).
- The user authentication (uauth) table.
- 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 ASA 1000V, there is a best-effort attempt to re-establish a TCP state.
- DHCP server address leases.
- State information for modules.

Auto Update Server Support in Failover Configurations

You can use the Auto Update Server to deploy software images and configuration files to ASA 1000Vs 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 ASA 1000V in the failover pair. See the “Configuring Auto Update Support” section on page 32-7, for more information.

The following restrictions and behaviors apply to Auto Update Server support in failover configurations:

- 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 ASA 1000Vs will fail.
- Only the primary ASA 1000V will perform the call home to the Auto Update Server. The primary ASA 1000V must be in the active state to call home. If it is not, the ASA 1000V automatically fails over to the primary ASA 1000V.
- Only the primary ASA 1000V downloads the software image or configuration file. The software image or configuration is then copied to the secondary ASA 1000V.
- The interface MAC address is from the primary ASA 1000V.
- The configuration file stored on the Auto Update Server or HTTP server is for the primary ASA 1000V 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 ASA 1000Vs are synchronizing configurations, if the standby ASA 1000V is in the failed state for any reason, or if the failover link is down.

1. Both ASA 1000Vs exchange the platform and ASDM software checksum and version information.
2. The primary ASA 1000V contacts the Auto Update Server. If the primary ASA 1000V is not in the active state, the ASA 1000V first fails over to the primary ASA 1000V and then contacts the Auto Update Server.
3. The Auto Update Server replies with software checksum and URL information.
4. If the primary ASA 1000V determines that the platform image file needs to be updated for either the active or standby ASA 1000V, the following occurs:
   -a. The primary ASA 1000V retrieves the appropriate files from the HTTP server using the URL from the Auto Update Server.
b. The primary ASA 1000V copies the image to the standby ASA 1000V and then updates the image on itself.

c. If both ASA 1000Vs have new image, the secondary (standby) ASA 1000V is reloaded first.
   – If hitless upgrade can be performed when the secondary ASA 1000V boots, then the secondary ASA 1000V becomes the active ASA 1000V and the primary ASA 1000V reloaded. The primary ASA 1000V becomes the active ASA 1000V when it has finished loading.
   – If hitless upgrade cannot be performed when the standby ASA 1000V boots, then both ASA 1000Vs reload at the same time.

d. If only the secondary (standby) ASA 1000V has new image, then only the secondary ASA 1000V reloaded. The primary ASA 1000V waits until the secondary ASA 1000V finishes reloading.

e. If only the primary (active) ASA 1000V has new image, the secondary ASA 1000V becomes the active ASA 1000V, and the primary ASA 1000V reloaded.

f. The update process starts again at Step 1.

5. If the ASA 1000V determines that the ASDM image file needs to be updated for either the primary or secondary ASA 1000V, the following occurs:

a. The primary ASA 1000V retrieves the ASDM image file from the HTTP server using the URL provided by the Auto Update Server.

b. The primary ASA 1000V copies the ASDM image to the standby ASA 1000V, if needed.

c. The primary ASA 1000V updates the ASDM image on itself.

d. The update process starts again at Step 1.

6. If the primary ASA 1000V determines that the configuration needs to be updated, the following occurs:

a. The primary ASA 1000V retrieves the configuration from the using the specified URL.

b. The new configuration replaces the old configuration on both ASA 1000Vs 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.bin, checksum: 0x94bcd0261cc992ae710faf8d244cf32
   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
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
auto-update: Fover copyfile, seq = 4 type = 1, pseq = 2501, len = 1024
auto-update: Fover copyfile, seq = 4 type = 1, pseq = 3001, len = 1024
auto-update: Fover copyfile, seq = 4 type = 1, pseq = 3501, len = 1024
auto-update: Fover copyfile, seq = 4 type = 1, pseq = 4001, len = 1024
auto-update: Fover copyfile, seq = 4 type = 1, pseq = 4501, len = 1024
auto-update: Fover copyfile, seq = 4 type = 1, pseq = 5001, len = 1024
auto-update: Fover copyfile, seq = 4 type = 1, pseq = 5501, len = 1024
auto-update: Fover copyfile, seq = 4 type = 1, pseq = 6001, len = 1024
auto-update: Fover copyfile, seq = 4 type = 1, pseq = 6501, len = 1024
auto-update: Fover copyfile, seq = 4 type = 1, pseq = 7001, len = 1024
auto-update: Fover copyfile, seq = 4 type = 1, pseq = 7501, len = 1024
auto-update: Fover copyfile, seq = 4 type = 1, pseq = 8001, len = 1024
auto-update: Fover copyfile, seq = 4 type = 1, pseq = 8501, len = 1024
auto-update: Fover copyfile, seq = 4 type = 1, pseq = 9001, len = 1024
auto-update: Fover file copy waiting at clock tick 6129280
fover_parse: Rcvd file copy ack, ret = 0, seq = 4
auto-update: Fover filecopy returns value: 0 at clock tick 6150260, upd time 145980 msecs
Auto-update client: update img on active unit...
fover_parse: Rcvd image info from mate
auto-update: HA safe reload: reload active waiting with mate state: 20
auto-update: HA safe reload: reload active waiting with mate state: 20
auto-update: HA safe reload: reload active waiting with mate state: 20
auto-update: HA safe reload: reload active waiting with mate state: 20
auto-update: HA safe reload: reload active waiting with mate state: 20
auto-update: HA safe reload: reload active waiting with mate state: 20
auto-update: HA safe reload: reload active waiting with mate state: 20
auto-update: HA safe reload: reload active waiting with mate state: 20
auto-update: HA safe reload: reload active waiting with mate state: 20
auto-update: HA safe reload: reload active waiting with mate state: 20
auto-update: HA safe reload: reload active waiting with mate state: 20
auto-update: HA safe reload: reload active waiting with mate state: 20
auto-update: HA safe reload: reload active waiting with mate state: 20
Beginning configuration replication: Sending to mate.
avto-update: HA safe reload: reload active waiting with mate state: 50
auto-update: HA safe reload: reload active waiting with mate state: 50
auto-update: HA safe reload: reload active waiting with mate state: 80
Auto-update: HA safe reload: reload active unit at clock tick: 6266860
Auto-update client: Succeeded: Image, version: 0x6d091b43ce96243e29a62f2330139419

The following syslog message is generated if the Auto Update process fails:
%ASA-4-612002: Auto Update failed: file version: version reason: reason
The file is “image,” “asdm,” or “configuration,” depending on which update failed. The version is the version number of the update. And the reason is the reason the update failed.

Failover Health Monitoring

The ASA 1000V monitors each ASA 1000V for overall health and for interface health. See the following sections for more information about how the ASA 1000V performs tests to determine the state of each ASA 1000V:

- ASA 1000V Health Monitoring, page 3-13
- Interface Monitoring, page 3-13
ASA 1000V Health Monitoring

The ASA 1000V determines the health of the other ASA 1000V by monitoring the failover link. When an ASA 1000V does not receive three consecutive hello messages on the failover link, the ASA 1000V 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 ASA 1000V takes depends upon the response from the other ASA 1000V. See the following possible actions:

- If the ASA 1000V receives a response on the failover interface, then it does not fail over.
- If the ASA 1000V does not receive a response on the failover link, but it does receive a response on another interface, then the ASA 1000V does not failover. The failover link is marked as failed. You should restore the failover link as soon as possible because the ASA 1000V cannot fail over to the standby while the failover link is down.
- If the ASA 1000V does not receive a response on any interface, then the standby ASA 1000V switches to active mode and classifies the other ASA 1000V 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 ASA 1000V 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. You should monitor important interfaces.

When an ASA 1000V 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 ASA 1000V performs network tests. The purpose of these tests is to generate network traffic to determine which (if either) ASA 1000V has failed. At the start of each test, each ASA 1000V clears its received packet count for its interfaces. At the conclusion of each test, each ASA 1000V looks to see if it has received any traffic. If it has, the interface is considered operational. If one ASA 1000V receives traffic for a test and the other ASA 1000V does not, the ASA 1000V that received no traffic is considered failed. If neither ASA 1000V has received traffic, then the next test is used.

2. Network Activity test—A received network activity test. The ASA 1000V 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 ASA 1000V ARP cache for the two most recently acquired entries. One at a time, the ASA 1000V sends ARP requests to these machines, attempting to stimulate network traffic. After each request, the ASA 1000V 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 ASA 1000V 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 all network tests fail for an interface, but this interface on the other ASA 1000V 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 ASA 1000V 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 ASA 1000V returns to standby mode if the interface failure threshold is no longer met.
Prerequisites for Active/Standby Failover

Note
If a failed ASA 1000V 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 ASA 1000V will fail again.

Failover Messages

When a failover occurs, both ASA 1000Vs send out system messages. This section includes the following topics:
- Failover System Messages, page 3-14
- Debugging Messages, page 3-14
- SNMP, page 3-14

Failover System Messages

The ASA 1000V 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. To enable logging, see Chapter 29, “ Configuring Logging.”

Note
During switchover, failover logically shuts down and then bring up interfaces, generating syslog 411001 and 411002 messages. This activity is normal.

Debugging Messages

To see debugging 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 the 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 30, “ Configuring SNMP” for more information.

Prerequisites for Active/Standby Failover

Both ASA 1000Vs must be identical ASA 1000Vs that are connected to each other through a dedicated failover link and, optionally, a Stateful Failover link.
Guidelines and Limitations

- To receive packets from both ASA 1000Vs in a failover pair, standby IP addresses need to be configured on all interfaces.
- The standby IP addresses are used on the ASA 1000V that is currently the standby ASA 1000V, and they must be in the same subnet as the active IP address on the corresponding interface on the active ASA 1000V.
- If you change the console terminal pager settings on the active ASA 1000V in a failover pair, the active console terminal pager settings change, but the standby ASA 1000V settings do not. A default configuration issued on the active ASA 1000V does affect behavior on the standby ASA 1000V.
- When you enable interface monitoring, you can monitor up to 250 interfaces on an ASA 1000V.
- By default, the ASA 1000V 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 on system performance.

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 3-15
- Configuring the Primary ASA 1000V, page 3-16
- Configuring the Secondary ASA 1000V, page 3-18
- Configuring Optional Active/Standby Failover Settings, page 3-19

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 ASA 1000V, as shown in the “Configuring the Primary ASA 1000V” section on page 3-16.</td>
</tr>
<tr>
<td>Step 2</td>
<td>Configure the secondary ASA 1000V, as shown in the “Configuring the Secondary ASA 1000V” section on page 3-18.</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 3-19.</td>
</tr>
</tbody>
</table>
## Configuring the Primary ASA 1000V

Follow the steps in this section to configure the primary ASA 1000V in a LAN-based, Active/Standby failover configuration. These steps provide the minimum configuration needed to enable failover on the primary ASA 1000V.

### 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 4, “Configuring Interfaces.”

### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> <code>failover lan unit primary</code></td>
<td>Designates the ASA 1000V as the primary ASA 1000V.</td>
</tr>
<tr>
<td><strong>Step 2</strong> <code>failover lan interface if_name interface_id</code></td>
<td>Specifies the interface to be used as the failover interface. This interface should not be used for any other purpose (except, optionally, the Stateful Failover link). The <code>if_name</code> argument assigns a name to the interface specified by the <code>interface_id</code> argument. The interface ID is the Ethernet interface.</td>
</tr>
<tr>
<td>Example: hostname(config)# failover lan interface folink gigabitEthernet0/2</td>
<td>Assigns the active and standby IP addresses to the failover link. 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 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 ASA 1000V, while the standby IP address stays with the secondary ASA 1000V.</td>
</tr>
<tr>
<td><strong>Step 3</strong> <code>failover interface ip if_name [ip_address mask standby ip_address]</code></td>
<td>Enables the interface.</td>
</tr>
<tr>
<td>Example: 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><strong>Step 4</strong> <code>interface interface_id no shutdown</code></td>
<td></td>
</tr>
<tr>
<td>Example: hostname(config)# interface gigabitEthernet0/2 hostname(config-if)# no shutdown</td>
<td></td>
</tr>
<tr>
<td>Command</td>
<td>Purpose</td>
</tr>
<tr>
<td>---------</td>
<td>---------</td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td>Configuring Active/Standby Failover Link</td>
</tr>
<tr>
<td><code>failover link if_name interface_id</code></td>
<td>(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).</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>hostname(config)# failover link statelink gigabitEthernet0/2</code></td>
<td>The <code>if_name</code> argument assigns a logical name to the interface specified by the <code>interface_id</code> argument. The <code>interface_id</code> argument can be the physical port name, such as Ethernet1, or a previously created subinterface, such as Ethernet0/2.3. This interface is an Ethernet interface. The failover configuration uses the interface ID (for example, gigabitEthernet0/2) instead of the interface name.</td>
</tr>
<tr>
<td><strong>Step 6</strong></td>
<td>Configuring Active/Standby IP Addresses</td>
</tr>
<tr>
<td><code>failover interface ip if_name [ip_address mask standby ip_address]</code></td>
<td>(Optional) Assigns an active and standby IP address to the Stateful Failover link.</td>
</tr>
<tr>
<td><strong>Example:</strong></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 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>
</tr>
<tr>
<td><strong>Step 7</strong></td>
<td>Enabling the Interface</td>
</tr>
<tr>
<td><code>interface interface_id no shutdown</code></td>
<td>(Optional) Enables the interface.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>hostname(config)# interface gigabitEthernet0/2</code></td>
<td></td>
</tr>
<tr>
<td><code>hostname(config-if)# no shutdown</code></td>
<td>If the Stateful Failover link uses the failover link or a data interface, skip this step. You have already enabled the interface.</td>
</tr>
<tr>
<td><strong>Step 8</strong></td>
<td>Enabling Failover</td>
</tr>
<tr>
<td><code>failover</code></td>
<td>Enables failover.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>hostname(config)# failover</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 9</strong></td>
<td>Saving the System Configuration</td>
</tr>
<tr>
<td><code>copy running-config startup-config</code></td>
<td>Saves the system configuration to flash memory.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>hostname(config)# copy running-config startup-config</code></td>
<td></td>
</tr>
</tbody>
</table>
Configuring the Secondary ASA 1000V

The only configuration required on the secondary ASA 1000V is for the failover interface. The secondary ASA 1000V requires these commands to communicate initially with the primary unit. After the primary ASA 1000V sends its configuration to the secondary ASA 1000V, the only permanent difference between the two configurations is the failover lan unit command, which identifies each ASA 1000V 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 ASA 1000V, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| Step 1  | **failover lan interface** *if_name*
|         | *interface_id* |
|         | Specifies the interface to be used as the failover interface. (Use the same settings that you used for the primary ASA 1000V.) |
|         | The *if_name* argument assigns a name to the interface specified by the *interface_id* argument. |
|         | This interface is an Ethernet interface. The failover configuration uses the interface ID (for example, gigabitEthernet0/2) instead of the interface name. |
| Example: | hostname(config)# failover lan interface folink gigabitEthernet0/2 |
| Step 2  | **failover interface ip** *if_name* [ip_address mask standby ip_address] |
|         | Assigns the active and standby IP addresses to the failover link. |
|         | To receive packets from both ASA 1000V in a failover pair, standby IP addresses need to be configured on all interfaces. |
| Note    | Enter this command exactly as you entered it on the primary ASA 1000V when you configured the failover interface on the primary ASA 1000V (including the same IP address). |
| Example: | hostname(config)# failover interface ip folink 172.27.48.1 255.255.255.0 standby 172.27.48.2 |
| Step 3  | **interface** *interface_id*
|         | no shutdown |
|         | Enables the interface. |
| Example: | hostname(config)# interface gigabitEthernet0/2
|         | hostname(config-if)# no shutdown |
| Step 4  | **failover lan unit secondary** |
|         | (Optional) Designates this ASA 1000V as the secondary ASA 1000V: |
| Example: | hostname(config)# failover lan unit secondary |
| Note    | This step is optional because, by default, ASA 1000Vs are designated as secondary unless previously configured. |
Configuring Optional Active/Standby Failover Settings

This section includes the following topics:

- Enabling HTTP Replication with Stateful Failover, page 3-19
- Disabling and Enabling Interface Monitoring, page 3-20
- Configuring Failover Criteria, page 3-20
- Configuring the ASA 1000V and Interface Health Poll Times, page 3-21
- Configuring Virtual MAC Addresses, page 3-21

You can configure the optional Active/Standby failover settings when initially configuring the primary ASA 1000V in a failover pair (see Configuring the Primary ASA 1000V, page 3-16) or on the active ASA 1000V 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:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>failover replication http</td>
<td>Enables HTTP state replication.</td>
</tr>
</tbody>
</table>

Example:

```
hostname (config)# failover replication http
```

Step 5

- **failover**

Example:

```
hostname(config)# failover
```

Enables failover. After you enable failover, the active ASA 1000V sends the configuration in running memory to the standby ASA 1000V. As the configuration synchronizes, the messages “Beginning configuration replication: Sending to mate” and “End Configuration Replication to mate” appear on the active ASA 1000V console.

Step 6

- **copy running-config startup-config**

Example:

```
hostname(config)# copy running-config startup-config
```

Saves the configuration to flash memory. Enter the command after the running configuration has completed replication.
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 256 interfaces on an ASA 1000V. By default, monitoring Ethernet interfaces is enabled and monitoring subinterfaces is disabled.

Hello messages are exchanged during every interface poll frequency time period between the ASA 1000V 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 ASA 1000Vs in single configuration mode, enter one of the following commands.

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>no monitor-interface if_name</code></td>
<td>Disables health monitoring for an interface.</td>
</tr>
<tr>
<td><code>monitor-interface if_name</code></td>
<td>Enables health monitoring for an interface.</td>
</tr>
</tbody>
</table>

### 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:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>failover interface-policy num[%]</code></td>
<td>Changes the default failover criteria.</td>
</tr>
<tr>
<td><strong>Example:</strong> hostname (config)# failover interface-policy 20%</td>
<td>When specifying a specific number of interfaces, the <code>num</code> argument can be from 1 to 256. When specifying a percentage of interfaces, the <code>num</code> argument can be from 1 to 100.</td>
</tr>
</tbody>
</table>
Configuring the ASA 1000V and Interface Health Poll Times

The ASA 1000V sends hello packets out of each data interface to monitor interface health. The ASA 1000V sends hello messages across the failover link to monitor ASA 1000V health. If the ASA 1000V does not receive a hello packet from the corresponding interface on the peer ASA 1000V 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 ASA 1000V to detect and respond to interface failures more quickly, but may consume more system resources. Increasing the poll and hold times prevents the ASA 1000V from failing over on networks with higher latency.

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>failover polltime interface [msec] time</td>
<td>Changes the interface poll and hold times. Valid values for 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>
<tr>
<td>Example: hostname (config): failover polltime interface msec 500 holdtime 5</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>failover polltime [unit] [msec] poll_time</td>
<td>Changes the unit poll and hold times. You cannot enter a holdtime value that is less than 3 times the unit poll time. With a faster poll time, the ASA 1000V can detect failure and trigger failover faster. However, faster detection can cause unnecessary switchovers when the network is temporarily congested.</td>
</tr>
<tr>
<td>Example: hostname(config)# failover polltime unit msec 200 holdtime msec 800</td>
<td></td>
</tr>
</tbody>
</table>

Configuring Virtual MAC Addresses

In Active/Standby failover, the MAC addresses for the primary ASA 1000V are always associated with the active IP addresses. If the secondary ASA 1000V boots first and becomes active, it uses the burned-in MAC address for its interfaces. When the primary ASA 1000V comes online, the secondary ASA 1000V obtains the MAC addresses from the primary ASA 1000V. The change can disrupt network traffic.

You can configure virtual MAC addresses for each interface to ensure that the secondary ASA 1000V uses the correct MAC addresses when it is the active ASA 1000V, even if it comes online before the primary ASA 1000V. If you do not specify virtual MAC addresses, the failover pair uses the burned-in NIC addresses as the MAC addresses.
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 ASA 1000V:

```
hostname (config): failover mac address phy_if active_mac standby_mac
```

**Example:**
```
hostname (config): failover mac address Ethernet0/2 00a0.c969.87c8 00a0.c918.95d8
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>failover mac address   </code></td>
<td>Configures the virtual MAC address for an interface.</td>
</tr>
<tr>
<td><code>phy_if active_mac      </code></td>
<td></td>
</tr>
<tr>
<td><code>standby_mac</code></td>
<td></td>
</tr>
</tbody>
</table>

The `phy_if` argument is the name of the interface, such as Ethernet1. The `active_mac` and `standby_mac` 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 `active_mac` address is associated with the active IP address for the interface, and the `standby_mac` is associated with the standby IP address for the interface.

There are multiple ways to configure virtual MAC addresses on the ASA 1000V. When more than one method has been used to configure virtual MAC addresses, the ASA 1000V 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 generated address.
3. The `failover mac address` command address.
4. The burned-in MAC address.

Use the `show interface` command to display the MAC address used by an interface.

## Controlling Failover

This sections describes how to control and monitor failover and includes the following topics:

- Forcing Failover, page 3-23
- Disabling Failover, page 3-23
- Restoring a Failed ASA 1000V, page 3-23
Forcing Failover

To force the standby ASA 1000V 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 ASA 1000V in a failover pair. The standby ASA 1000V becomes the active ASA 1000V.</td>
</tr>
<tr>
<td>no failover active</td>
<td>Forces a failover when entered on the active ASA 1000V in a failover pair. The active ASA 1000V becomes the standby ASA 1000V.</td>
</tr>
</tbody>
</table>

<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 ASA 1000V to be maintained until you restart. For example, the standby ASA 1000V remains in standby mode so that both ASA 1000Vs do not start passing traffic. To make the standby ASA 1000V active (even with failover disabled), see the “Forcing Failover” section on page 3-23.</td>
</tr>
</tbody>
</table>

Restoring a Failed ASA 1000V

To restore a failed ASA 1000V 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 ASA 1000V to an unfailed state. Restoring a failed ASA 1000V to an unfailed state does not automatically make it active; restored ASA 1000Vs remain in the standby state until made active by failover (forced or natural).</td>
</tr>
</tbody>
</table>

Testing the Failover Functionality

To test failover functionality, perform the following steps:

Step 1  Test that the active ASA 1000V 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 ASA 1000V:
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 ASA 1000V to active status by enter the following command on the newly active ASA 1000V:

hostname(config)# no failover active

---

## Monitoring Active/Standby Failover

**Note**

After a failover event, you should either restart ASDM or switch to another device listed in the Devices pane, then return to the original ASA 1000V 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 ASA 1000V.</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, see the command reference.

## Feature History for Active/Standby Failover

Table 3-2 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>Failover support for the ASA 1000V</td>
<td>8.7(1)</td>
<td>The interface assigned to active/standby failover pairs for the ASA 1000V is GigabitEthernet0/2.</td>
</tr>
</tbody>
</table>
Configuring Interfaces

This chapter describes tasks to complete the interface configuration and includes the following sections:

- Information About Interfaces, page 4-1
- Guidelines and Limitations, page 4-4
- Default Settings, page 4-4
- Configuring Communication with VNMC, page 4-5
- Configuring Interfaces, page 4-6
- Monitoring Interfaces, page 4-10
- Feature History for Interfaces, page 4-11

Information About Interfaces

- Ethernet Interfaces, page 4-1
- Security Profile Interfaces, page 4-2
- How to Apply the Security Policy to Interfaces, page 4-3
- To-the-Box Traffic, page 4-3
- vPath Tagging, page 4-3
- Interface Security Levels, page 4-4

Ethernet Interfaces

When you initially provision the ASA 1000V, you associate the ASA 1000V Ethernet interfaces with port groups that correspond to the Nexus 1000V port profiles for these interfaces. Nexus 1000V port profiles associate the interface to a VLAN, in addition to specifying other switch parameters to the interface. Assigning the same port profile to multiple interfaces has the effect of applying the same switchport configuration to these interfaces. For more information about configuring port profiles, see the Nexus 1000V documentation.

Each ASA 1000V provides four available Ethernet interfaces for data and failover traffic: one for management, 2 for through traffic, and one for a failover link.
• Management 0/0—For management-only traffic, named management, with IP address parameters you specified when you deployed the ASA 1000V. You can change these parameters using this chapter if desired, but the name is fixed.

• GigabitEthernet 0/0—Configure this interface as a data interface according to this chapter.

• GigabitEthernet 0/1—Configure this interface as a data interface according to this chapter.

• GigabitEthernet 0/2—For failover traffic, with IP address parameters you specified when you deployed the ASA 1000V. To change the failover link parameters, see Chapter 3, “Configuring Active/Standby Failover.”

Configure the two data interfaces as the inside (higher security level) interface and as the outside (lower security level) interface. See the “Interface Security Levels” section on page 4-4 for information about security levels.

**Security Profile Interfaces**

Security profile interfaces correspond to security profiles on the Nexus 1000V. On a given network, security profiles let you segregate a class of virtual machines (VMs) from other VMs; for example, web servers from application servers. Security profiles let you apply a security policy based on a class of VMs instead of based on IP addresses.

• Security Profile Coordination with VNMC, page 4-2

• Security Profiles on the Inside Interface, page 4-2

• Interface-Based Policy, page 4-2

• Port Profiles for Security Profile Interfaces, page 4-3

**Security Profile Coordination with VNMC**

When you create a security profile interface on the ASA 1000V, a security profile with the same name is added to Cisco VNMC automatically for use in port profiles on the Nexus 1000V.

**Security Profiles on the Inside Interface**

When traffic enters the ASA 1000V inside interface, the ASA 1000V can identify the security profile for the traffic based on a tag included in the packet (called vPath tagging; see the “vPath Tagging” section on page 4-3). The ASA 1000V can only receive traffic from security profiles on one Ethernet interface: the service interface, which must be the inside interface. The service interface is automatically made to be the inside interface. Traffic on the outside interface is untagged.

**Interface-Based Policy**

The ASA operating system has an interface-based security policy. Security profiles are treated as “interfaces” within the ASA to take advantage of the ASA interface-based policy. The security profile is a class of traffic that is a subset of all of the traffic sent or received on the inside interface.
Port Profiles for Security Profile Interfaces

Like Ethernet interfaces, security profile interfaces are also associated with Nexus 1000V port profiles. Just as a security profile is attached to the inside Ethernet interfaces, the Nexus 1000V port profile for the security profile must be on the same VLAN as the inside interface port profile.

How to Apply the Security Policy to Interfaces

A security policy determines the allowed behavior of traffic: whether packets are allowed from the outside to the inside; whether to perform NAT on inside networks; whether to apply inspection on traffic from the inside when it access a server on the outside, and so on. Depending on the feature, you may need to identify one or more interfaces on which to apply the feature.

All features that need to refer to the outside interface must refer to the outside Ethernet interface directly. Because security profiles are only attached to the inside interface (the service interface), all features applied to the inside interface must refer to the specific security profile, and not the inside interface directly. For security policy purposes, the inside interface is divided up into separate security profiles.

To-the-Box Traffic

To-the-box management traffic is received by the inside Ethernet interface, not a security-profile interface. Similarly, from-the-box traffic is sent from the inside interface.

vPath Tagging

The Nexus 1000V applies vPath tagging to traffic with a destination MAC address equal to the ASA 1000V inside interface.

If the ASA 1000V receives through traffic on the inside interface that are not tagged, then the ASA 1000V drops the packets (see the `show asp drop` command to view dropped packets).

Broadcast and multicast traffic is not tagged (ARP and DHCP). Broadcast and multicast traffic is handled by the inside Ethernet interface on the ASA 1000V, not security profile interfaces.
Interface Security Levels

Both Ethernet and security profile interfaces use security levels:

- The inside interface has a security level of 100 (highest). We recommend leaving this level as is.
- The security-profile interfaces have a security level of 0. You must change this level to a higher security level, for example 100.
- The outside interface has a security level of 0 (lowest). We recommend leaving this level as is.
- The management interface has a security level of 0. We recommend leaving this level as is.

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 cannot communicate between interfaces at the same security level.
- Inspection engines—Some application inspection engines are dependent on the security level.
  - 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 ASA.

Guidelines and Limitations

Failover Guidelines

To configure the failover link, see Chapter 3, “Configuring Active/Standby Failover.”

Additional Guidelines

- The Management 0/0 interface can only be configured as a management-only interface.
- You can create up to 256 security profile interfaces.
- The ASA 1000V supports jumbo Ethernet packets; configure the MTU and TCP maximum segment size as desired.

Default Settings

Default Security Level

- The inside interface has a security level of 100 (highest). We recommend leaving this level as is.
- The security-profile interfaces have a security level of 0. You must change this level to a higher security level, for example 100.
- The outside interface has a security level of 0 (lowest). We recommend leaving this level as is.
- The management interface has a security level of 0. We recommend leaving this level as is.

Default State of Interfaces

- Ethernet interfaces—Disabled.
- Management interface—Enabled as part of ASA 1000V deployment.
Security profile interfaces—Disabled.

Default Speed and Duplex
By default, the speed and duplex for Ethernet interfaces are set to auto-negotiate.

Default MAC Addresses
By default, the Ethernet interfaces use a MAC address dynamically assigned when you deployed the ASA 1000V. All associated security profile interfaces use the same MAC address.

## Configuring Communication with VNMC

You must enable communication with VNMC before you can set up security profile interfaces.

### 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>vnmc policy-agent</td>
<td>Enables the ASA 1000V to communicate with the VNMC and enters VNMC policy-agent configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname (config)# vnmc policy-agent</td>
<td></td>
</tr>
</tbody>
</table>

| **Step 2**               |                                                                          |
| login username username password | Specifies your VNMC login credentials. Your account must have administrator privilege on VNMC. |
| Example:                  |                                                                          |
| hostname(config-vnmc-policy-agent)# login username exampleuser1 password cisco123 |                                                                 |

| **Step 3**               |                                                                          |
| shared-secret shared-secret | Specifies the shared secret for encryption of the ASA 1000V connection to the VNMC. The shared secret is hidden for security purposes. Both the VNMC and the ASA 1000V share the same keys. When a managed endpoint (the ASA 1000V) sends the request to the VNMC, the ASA 1000V includes a hash value for authentication. The VNMC can use the same hash generation algorithm to check whether or not the request is from a trusted source. Because the VNMC IP address is used together with the shared secret to generate the hash value, the VNMC IP address must be the real IP address of the VNMC (not changed through NAT); otherwise, the hash verification of the IP address fails on the VNMC. NAT between the ASA 1000V and VNMC is not supported. |
| Example:                  |                                                                          |
| hostname(config-vnmc-policy-agent)# shared-secret adamyauchrip |                                                                 |

Both the VNMC and the ASA 1000V share the same keys. When a managed endpoint (the ASA 1000V) sends the request to the VNMC, the ASA 1000V includes a hash value for authentication. The VNMC can use the same hash generation algorithm to check whether or not the request is from a trusted source. Because the VNMC IP address is used together with the shared secret to generate the hash value, the VNMC IP address must be the real IP address of the VNMC (not changed through NAT); otherwise, the hash verification of the IP address fails on the VNMC. NAT between the ASA 1000V and VNMC is not supported.
Configuring Interfaces

This section includes the following topics:

- Configuring the Inside and Outside Ethernet Interfaces, page 4-6
- Configuring Security Profile Interfaces, page 4-8
- Associating Security Profile Interfaces with an Ethernet Interface, page 4-9
- Setting the vPath MTU, page 4-9

Configuring the Inside and Outside Ethernet Interfaces

This section describes how to set the name, IPv4 address, and other options for the inside and outside interfaces.

If you want to change parameters for the management interface, you can also use this procedure.

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. To configure the failover and state links, see Chapter 3, “Configuring Active/Standby Failover.”

---

### Command Purpose

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 4</strong> registration host <em>ip_address</em></td>
<td>Specifies the IP address or hostname of the host on which the VNMC is running.</td>
</tr>
</tbody>
</table>
| *Example:*  
  hostname(config-vnmc-policy-agent)# registration host 192.168.1.1 | |
| **Step 5** vnmc org *org-path* | Configures the organization path. You must configure the organization hierarchy on the ASA 1000V to match the organization hierarchy that has been configured on the VSM. The organization path must be in the following format:  
  root/name_of_tenant/name_of_datacenter/name_of_application/name_of_tier  
  A maximum of four layers is allowed, and the organization path must be located under the root directory.  
  Each ASA 1000V must specify a unique organization path. There must not be another ASA 1000V assigned at this ORG path in VNMC.  
  **Note** Make sure that the VNMC policy agent has been started and registered before continuing to the next procedure. |
| *Example:*  
  hostname(config)# vnmc org  
  root/tenant1/datacenter1/application1/tier1 | |

---

---

---

---

---

---
## Detailed Steps

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>`interface gigabitethernet 0/0</td>
<td>1`</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td><code>nameif name</code></td>
<td>Names the interface.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Do one of the following:</td>
<td>Sets the IP address manually.</td>
</tr>
<tr>
<td></td>
<td><code>ip address ip_address [mask] [standby ip_address]</code></td>
<td><strong>Note</strong> For use with failover, you must set the IP address and standby address manually; DHCP is not supported.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>ip address dhcp [setroute]</code></td>
<td>Obtains an IP address from a DHCP server.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>(Optional)</td>
<td>Sets the speed. The default setting is <code>auto</code>. The <code>speed nonegotiate</code> command disables link negotiation.</td>
</tr>
<tr>
<td></td>
<td>`speed {auto</td>
<td>10</td>
</tr>
<tr>
<td>5</td>
<td>(Optional)</td>
<td>Sets the duplex. The <code>auto</code> setting is the default.</td>
</tr>
<tr>
<td></td>
<td>`duplex {auto</td>
<td>full</td>
</tr>
</tbody>
</table>
Step 6  (Optional)

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>mac-address mac_address</td>
<td>Assigns a private MAC address to this interface. The mac_address</td>
</tr>
<tr>
<td>[standby mac_address]</td>
<td>argument is in H.H.H format, where H is a 16-bit hexadecimal digit.</td>
</tr>
<tr>
<td>Example:</td>
<td>For example, the MAC address 00-0C-F1-42-4C-DE is entered as 000C.F142.4CDE.</td>
</tr>
<tr>
<td>hostname(config-if)#</td>
<td>The first two bytes of a manual MAC address cannot be A2 if you</td>
</tr>
<tr>
<td>mac-address 000C.F142.4CDE</td>
<td>also want to use auto-generated MAC addresses.</td>
</tr>
<tr>
<td></td>
<td>For use with failover, set the standby MAC address. If the active</td>
</tr>
<tr>
<td></td>
<td>ASA 1000V fails over and the standby ASA 1000V becomes</td>
</tr>
<tr>
<td></td>
<td>active, the new active ASA 1000V starts using the active MAC</td>
</tr>
<tr>
<td></td>
<td>addresses to minimize network disruption, while the old active ASA</td>
</tr>
<tr>
<td></td>
<td>1000V uses the standby address.</td>
</tr>
</tbody>
</table>

Step 7  

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>no shutdown</td>
<td>Enables the interface. To disable the interface, enter the shutdown</td>
</tr>
<tr>
<td>Example:</td>
<td>command.</td>
</tr>
<tr>
<td>hostname(config-if)#</td>
<td>Sets the maximum transmission unit (MTU) for normal or jumbo</td>
</tr>
<tr>
<td>no shutdown</td>
<td>Ethernet packets, between 64 and 9216 bytes. The default is 1500</td>
</tr>
<tr>
<td></td>
<td>bytes. For jumbo packets, set this value to be higher, for example</td>
</tr>
<tr>
<td></td>
<td>9000. You cannot set this value to be higher than the vPath MTU</td>
</tr>
<tr>
<td></td>
<td>that you set in the “Setting the vPath MTU” section on page 4-9.</td>
</tr>
<tr>
<td></td>
<td>For optimal performance, set the interface MTU to a maximum of the</td>
</tr>
<tr>
<td></td>
<td>vPath MTU minus 164 bytes (twice the size of the maximum vPath header,</td>
</tr>
<tr>
<td></td>
<td>which is 82 bytes). Make sure you specify the interface ID, not the</td>
</tr>
<tr>
<td></td>
<td>interface name.</td>
</tr>
<tr>
<td>hostname(config-if)#</td>
<td>Sets the maximum transmission unit (MTU) for normal or jumbo</td>
</tr>
<tr>
<td>mtu gigabitethernet0/1 9200</td>
<td>Ethernet packets, between 64 and 9216 bytes. The default is 1500</td>
</tr>
<tr>
<td></td>
<td>bytes. For jumbo packets, set this value to be higher, for example</td>
</tr>
<tr>
<td></td>
<td>9000. You cannot set this value to be higher than the vPath MTU</td>
</tr>
<tr>
<td></td>
<td>that you set in the “Setting the vPath MTU” section on page 4-9.</td>
</tr>
<tr>
<td></td>
<td>For optimal performance, set the interface MTU to a maximum of the</td>
</tr>
<tr>
<td></td>
<td>vPath MTU minus 164 bytes (twice the size of the maximum vPath header,</td>
</tr>
<tr>
<td></td>
<td>which is 82 bytes). Make sure you specify the interface ID, not the</td>
</tr>
<tr>
<td></td>
<td>interface name.</td>
</tr>
</tbody>
</table>

Step 9  

To configure the outside interface, repeat these steps for the other available interface.

### Configuring Security Profile Interfaces

To configure a security profile interface, perform the following steps.

**Detailed Steps**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 interface security-profile number</td>
<td>Configures a security profile interface and enters interface configuration mode. The number argument specifies the security profile interface ID, between 1 and 256.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname (config)# interface security-profile1</td>
<td></td>
</tr>
<tr>
<td>Step 2 nameif interface_name</td>
<td>Names the interface. This name is only used within the ASA 1000V configuration. 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></td>
</tr>
<tr>
<td>hostname (config-if)# nameif profile1-ifc</td>
<td></td>
</tr>
</tbody>
</table>
Chapter 4      Configuring Interfaces

Configuring Interfaces

Associating Security Profile Interfaces with an Ethernet Interface

You must associate all security profiles with the inside interface by identifying the inside interface as the service interface.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>service-interface security-profile all inside_interface_name</code></td>
<td>Associates the security profile interfaces with the inside Ethernet interface.</td>
</tr>
</tbody>
</table>

Example:

```
hostname(config-if)# service-interface security-profile all inside
```

Setting the vPath MTU

The ASA 1000V receives encapsulated packets from virtual machines (VMs) using a packet-redirection mechanism known as vPath (see the “vPath Tagging” section on page 4-3). Due to the size of these vPath headers (up to 82 bytes), it is possible for a payload to require fragmentation after the vPath header has been added. The ASA 1000V has the ability to transparently handle this overhead without requiring the VMs to reduce their MTU to account for these additional bytes. The ASA 1000V can split a packet exceeding the uplink MTU into two vPath fragments when adding the vPath encapsulation before sending the fragments over Ethernet. The vPath fragments are reassembled by the Virtual Ethernet Module (VEM) in the Nexus 1000V switch before the packet is delivered to the destination VM.

---

**Step 3**

```
security-profile security_profile_name
```

Example:

```
hostname(config-if)# security-profile test-profile-1
```

Specifies the name of the security profile. The `security-profile_name` argument can range from 1 to 256 characters. When you add a security profile interface, this name is used to create the security profile in VNMC.

You cannot associate the same security profile name with two different security profile interfaces. An error message occurs for this type of configuration.

You cannot change the configured security profile name until the old security profile name is removed. Whenever the security profile is removed on an interface, all connections based on that security profile interface are cleared.

**Step 4**

```
no shutdown
```

Example:

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

Enables the interface. To disable the interface, enter the `shutdown` command.

---

**Command Purpose**

- `service-interface security-profile all inside_interface_name`: Associates the security profile interfaces with the inside Ethernet interface.
The vPath MTU setting configures how the vPath module in the ASA 1000V fragments a packet so that it complies with the MTU on the path from the ASA 1000V to the destination VM. The vPath module operates below the IP layer on the ASA 1000V and is therefore independent of IP fragmentation (see the “Configuring the Inside and Outside Ethernet Interfaces” section on page 4-6). The VEM and vPath module on the ASA 1000V work together to present a valid IP datagram (fragment or otherwise) to the VMs and to the ASA 1000V. The ASA 1000V enforces a TCP MSS setting that already accounts for the additional overhead for vPath.

There may be other encapsulations in the path between the ASA 1000V and the VM. For example, if VXLAN is used between the ASA 1000V and the VM, then 50 additional bytes are used for the packets.

To avoid vPath fragmentation when additional overhead is present, do one of the following:

- Decrease the vPath MTU to accommodate the VXLAN encapsulation (50 bytes). The default value of the vPath MTU is 9000 bytes, which matches the uplink port default MTU on the Nexus 1000V. For example, set the vPath MTU to 8950.
- Increase the uplink MTUs to avoid any vPath fragmentation and allow VXLAN encapsulation. To accommodate VXLAN encapsulation, you could increase the Nexus 1000V MTU to 9050.
- Reduce the MTU setting on the VMs to account for the additional overhead.

### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>vpath path-mtu bytes</code></td>
<td>Configures the vPath MTU threshold. The <code>bytes</code> argument defines the MTU that can be sent on the physical uplink interface to the Nexus 1000V switch, between 64 and 65535. The default is 9000 bytes. The MTU should be at least 164 bytes (twice the size of the maximum vPath header, which is a maximum of 82 bytes).</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><code>show interface</code></td>
<td>Displays interface statistics.</td>
</tr>
<tr>
<td><code>show interface ip brief</code></td>
<td>Displays interface IP addresses and status.</td>
</tr>
<tr>
<td><code>show interface security-profile</code></td>
<td>Displays the runtime status and security profile interface statistics.</td>
</tr>
<tr>
<td><code>show running-config interface</code></td>
<td>Displays the interface statistics for the current running configuration.</td>
</tr>
<tr>
<td><code>show running-config mtu</code></td>
<td>Displays the MTU for the current running configuration.</td>
</tr>
<tr>
<td><code>show running-config vnmc policy-agent</code></td>
<td>Displays the running configuration for the VNMC policy agent.</td>
</tr>
<tr>
<td><code>show vm</code></td>
<td>Displays the recommended values for memory and CPU resources and the actual CPU resource usage in real-time by the running VM.</td>
</tr>
<tr>
<td><code>show vnmc policy-agent</code></td>
<td>Shows the status of the VNMC policy agent.</td>
</tr>
</tbody>
</table>
### Feature History for Interfaces

Table 4-1 lists the release history.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Security profile interfaces</td>
<td>8.7(1)</td>
<td>We introduced security profile interfaces. Security profile interfaces correspond to security profiles on the Nexus 1000V. On a given network, security profiles let you segregate a class of virtual machines (VMs) from other VMs; for example, web servers from application servers. Security profiles let you apply a security policy based on a class of VMs instead of based on IP addresses. We introduced or modified the following commands: <code>interface security-profile</code>, <code>security-profile mtu</code>, <code>vpath path-mtu</code>, <code>clear interface security-profile</code>, <code>clear configure interface security-profile</code>, <code>show interface security-profile</code>, <code>show running-config interface security-profile</code>, <code>show interface ip brief</code>, <code>show running-config mtu</code>, <code>show vsn ip binding</code>, <code>show vsn security-profile</code>.</td>
</tr>
<tr>
<td>Service interface</td>
<td>8.7(1)</td>
<td>The service interface is the Ethernet interface associated with security profile interfaces. You can only configure one service interface, which must be the inside interface. We introduced the following command: <code>service-interface security-profile all</code>.</td>
</tr>
<tr>
<td>VNMC policy agent</td>
<td>8.7(1)</td>
<td>The VNMC policy agent enables policy configuration through both the ASDM and VNMC modes. It includes a web server that receives XML-based requests from Cisco VNMC over HTTPS and converts it to the ASA 1000V configuration. We introduced the following commands: <code>vnmc policy-agent</code>, <code>login</code>, <code>shared-secret</code>, <code>registration host</code>, <code>vnmc org</code>, <code>show vnmc policy-agent</code>, <code>show running-config vnmc policy-agent</code>, <code>clear configure vnmc policy-agent</code>.</td>
</tr>
</tbody>
</table>
CHAPTER 5

Configuring the Hostname, Domain Name, Passwords, and Other Basic Settings

This chapter describes how to configure basic settings on your ASA 1000V that are typically required for a functioning configuration and includes the following sections:

- Configuring the Hostname, Domain Name, and Passwords, page 5-1
- Setting the Date and Time, page 5-3
- Configuring the Master Passphrase, page 5-6
- Configuring the DNS Server, page 5-10

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 5-2
- Changing the Enable Password, page 5-2
- Setting the Hostname, page 5-3
- Setting the Domain Name, page 5-3
Changing the Login Password

To change the login password, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>(passwd</td>
<td>password) password</td>
</tr>
</tbody>
</table>

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>enable password password</td>
<td>Changes the enable password, which lets you enter privileged EXEC mode. By default, the enable password is blank. 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.</td>
</tr>
</tbody>
</table>

Example:

hostname(config)# passwd Pa$$w0rd
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 ASA 1000V. 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 ASA 1000V, 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.</td>
</tr>
</tbody>
</table>

Example:

```
hostname(config)# hostname farscape
farscape(config)#
```

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><code>domain-name name</code></td>
<td>Specifies the domain name for the ASA 1000V. The ASA 1000V appends the domain name as a suffix to unqualified names. For example, if you set the domain name to &quot;example.com,&quot; and specify a syslog server by the unqualified name of &quot;jupiter,&quot; then the ASA 1000V qualifies the name to &quot;jupiter.example.com.&quot; The default domain name is default.domain.invalid.</td>
</tr>
</tbody>
</table>

Example:

```
hostname(config)# domain-name example.com
```

Setting the Date and Time

This section includes the following topics:

- Setting the Time Zone and Daylight Saving Time Date Range, page 5-4
- Setting the Date and Time Using an NTP Server, page 5-5
- Setting the Date and Time Manually, page 5-6
## Setting the Time Zone and Daylight Saving Time Date Range

To change the time zone and daylight saving time date range, 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>clock timezone zone [-]hours [minutes]</code></td>
<td>Sets the time zone. By default, the time zone is UTC and the daylight saving time date range is from 2:00 a.m. on the first Sunday in April to 2:00 a.m. on the last Sunday in October. Where <code>zone</code> specifies the time zone as a string, for example, PST for Pacific Standard Time. The <code>[-]hours</code> value sets the number of hours of offset from UTC. For example, PST is -8 hours. The <code>minutes</code> value sets the number of minutes of offset from UTC.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>hostname(config)# clock timezone PST -8</code></td>
<td></td>
</tr>
</tbody>
</table>

| **Step 2** | To change the date range for daylight saving time from the default, enter one of the following commands. The default recurring date range is from 2:00 a.m. on the second Sunday in March to 2:00 a.m. on the first Sunday in November. | |
| `clock summer-time zone date {day month | month day} year hh:mm {day month | month day} year hh:mm [offset]` | Sets the start and end dates for daylight saving time as a specific date in a specific year. If you use this command, you need to reset the dates every year. The `zone` value specifies the time zone as a string, for example, PDT for Pacific Daylight Time. The `day` value sets the day of the month, from 1 to 31. You can enter the day and month as April 1 or as 1 April, depending on your standard date format. The `month` value sets the month as a string. You can enter the day and month as April 1 or as 1 April, depending on your standard date format. The `year` value sets the year using four digits, for example, 2004. The year range is 1993 to 2035. The `hh:mm` value sets the hour and minutes in 24-hour time. The `offset` value sets the number of minutes to change the time for daylight saving time. By default, the value is 60 minutes. |
| **Example:** | | |
| `hostname(config)# clock summer-time PDT 1 April 2010 2:00 60` | | |

| `clock summer-time zone recurring [week weekday month hh:mm week weekday month hh:mm] [offset]` | Specifies the start and end dates for daylight saving time, in the form of a day and time of the month, and not a specific date in a year. This command enables you to set a recurring date range that you do not need to change yearly. The `zone` value specifies the time zone as a string, for example, PDT for Pacific Daylight Time. The `week` value specifies the week of the month as an integer between 1 and 4 or as the words first or last. For example, if the day might fall in the partial fifth week, then specify last. The `weekday` value specifies the day of the week: Monday, Tuesday, Wednesday, and so on. The `month` value sets the month as a string. The `hh:mm` value sets the hour and minutes in 24-hour time. The `offset` value sets the number of minutes to change the time for daylight savings time. By default, the value is 60 minutes. |
| **Example:** | | |
| `hostname(config)# clock summer-time PDT recurring first Monday April 2:00 60` | | |
# Setting the Date and Time Using an NTP Server

To obtain the date and time from an NTP server, perform the following steps:

## Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>ntp authenticate</td>
</tr>
<tr>
<td><em>Example:</em></td>
<td>hostname(config)# ntp authenticate</td>
</tr>
<tr>
<td></td>
<td>Enables authentication with an NTP server.</td>
</tr>
<tr>
<td>Step 2</td>
<td>ntp trusted-key <em>key_id</em></td>
</tr>
<tr>
<td><em>Example:</em></td>
<td>hostname(config)# ntp trusted-key 1</td>
</tr>
<tr>
<td></td>
<td>Specifies an authentication key ID to be a trusted key, which is required for authentication with an NTP server.</td>
</tr>
<tr>
<td></td>
<td>The <em>key_id</em> argument is a value between 1 and 4294967295. You can enter multiple trusted keys for use with multiple servers.</td>
</tr>
</tbody>
</table>
| Step 3  | ntp authentication-key *key_id*  
|        | md5  
|        | *key*  |
| *Example:* | hostname(config)# ntp authentication-key 1 md5 aNiceKey  |
|   | Sets a key to authenticate with an NTP server. |
|      | The *key_id* argument is the ID you set in Step 2 using the ntp trusted-key command, and the *key* argument is a string up to 32 characters long. |
| Step 4  | ntp server *ip_address*  
|        | [key *key_id*]  
|        | [source *interface_name*]  
|        | [prefer]  |
| *Example:* | hostname(config)# ntp server 10.1.1.1 key 1 prefer  |
|   | Identifies an NTP server. |
|      | The *key_id* argument is the ID you set in Step 2 using the ntp trusted-key command. |
|      | The source *interface_name* keyword-argument pair identifies the outgoing Ethernet interface for NTP packets if you do not want to use the default Ethernet interface in the routing table. |
|      | The prefer keyword sets this NTP server as the preferred server if multiple servers have similar accuracy. NTP uses an algorithm to determine which server is the most accurate and synchronizes to that one. If servers are of similar accuracy, then the prefer keyword specifies which of those servers to use. However, if a server is significantly more accurate than the preferred one, the ASA 1000V uses the more accurate one. For example, the ASA 1000V uses a server of stratum 2 over a server of stratum 3 that is preferred. |
|      | You can identify multiple servers; the ASA 1000V uses the most accurate server. |
Setting the Date and Time Manually

To set the date and time manually, perform the following steps:

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`clock set hh:mm:ss (month day</td>
<td>day month)</td>
</tr>
<tr>
<td></td>
<td>The <code>hh:mm:ss</code> argument sets the hour, minutes, and seconds in 24-hour</td>
</tr>
<tr>
<td></td>
<td>time. For example, enter <strong>20:54:00</strong> for 8:54 pm.</td>
</tr>
<tr>
<td></td>
<td>The <code>day</code> value sets the day of the month, from 1 to 31. You can enter</td>
</tr>
<tr>
<td></td>
<td>the day and month as <strong>april 1</strong> or as <strong>1 april</strong>, for example, depending</td>
</tr>
<tr>
<td></td>
<td>on your standard date format.</td>
</tr>
<tr>
<td></td>
<td>The <code>month</code> value sets the month. Depending on your standard date format,</td>
</tr>
<tr>
<td></td>
<td>you can enter the day and month as <strong>april 1</strong> or as <strong>1 april</strong>.</td>
</tr>
<tr>
<td></td>
<td>The <code>year</code> value sets the year using four digits, for example, <strong>2004</strong>.</td>
</tr>
<tr>
<td></td>
<td>The year range is from 1993 to 2035.</td>
</tr>
<tr>
<td></td>
<td>The default time zone is UTC. If you change the time zone after you enter</td>
</tr>
<tr>
<td></td>
<td>the <code>clock set</code> command using the <code>clock timezone</code> command, the time</td>
</tr>
<tr>
<td></td>
<td>automatically adjusts to the new time zone.</td>
</tr>
<tr>
<td></td>
<td>This command sets the time in the hardware chip, and does not save the</td>
</tr>
<tr>
<td></td>
<td>time in the configuration file. This time endures reboots. Unlike the other</td>
</tr>
<tr>
<td></td>
<td><code>clock</code> commands, this command is a privileged EXEC command. To reset</td>
</tr>
<tr>
<td></td>
<td>the clock, you need to set a new time with the <code>clock set</code> command.</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 5-6
- Adding or Changing the Master Passphrase, page 5-7
- Disabling the Master Passphrase, page 5-9
- Recovering the Master Passphrase, page 5-9

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:

- IPsec site-to-site
- Failover
- AAA servers
- Logging
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>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><strong>key config-key password-encryption</strong> [new_passphrase [old_passphrase]]</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>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.</td>
</tr>
<tr>
<td>hostname(config)# key config-key password-encryption Old key: bumblebee New key: haverford Confirm key: haverford</td>
<td>If you do not enter the new passphrase in the command, you are prompted for it.</td>
</tr>
<tr>
<td></td>
<td>When you want to change the passphrase, you also have to enter the old passphrase.</td>
</tr>
<tr>
<td></td>
<td>See the “Examples” section on page 5-8 for examples of the interactive prompts.</td>
</tr>
<tr>
<td><strong>Note</strong></td>
<td>Use the interactive prompts to enter passwords to avoid having the passwords logged in the command history buffer.</td>
</tr>
<tr>
<td></td>
<td>Use the <strong>no key config-key password-encrypt</strong> command with caution, because it changes the encrypted passwords into plain text passwords. You can use the <strong>no</strong> form of this command when downgrading to a software version that does not support password encryption.</td>
</tr>
</tbody>
</table>
Chapter 5  Configuring the Hostname, Domain Name, Passwords, and Other Basic Settings

Configuring the Master Passphrase

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 2 password encryption aes</td>
<td>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.</td>
</tr>
</tbody>
</table>

Example:

```
hostname(config)# key config-key password-encryption 12345678
Old key: 12345678
```

Step 3 write memory

Example:

```
hostname(config)# 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.

Examples

In the following configuration example, no previous key is present:

```
hostname (config)# key config-key password-encryption
```

In the following configuration example, a key already exists:

```
hostname (config)# key config-key password-encryption 12345678
```

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 5-9.
- 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>Step 1</td>
<td></td>
</tr>
<tr>
<td>no key config-key password-encryption [old_passphrase]</td>
<td>Removes the master passphrase.</td>
</tr>
</tbody>
</table>

Example:

hostname(config)# no key config-key password-encryption

Warning! 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.

Old key: bumblebee

Step 2 | write memory |

Example:

hostname(config)# write memory

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.

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.
Configuring the DNS Server

Some ASA 1000V 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 ASA 1000V can resolve the name by communicating with a DNS server.

**Note**
The ASA 1000V 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.

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 7-1 for more information about routing.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Enables the ASA 1000V to send DNS requests to a DNS server to perform a name lookup for supported commands.</td>
</tr>
<tr>
<td><code>dns domain-lookup interface_name</code></td>
<td>The <code>interface_name</code> argument specifies the Ethernet interface name.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><code>hostname(config)# dns domain-lookup inside</code></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Specifies the DNS server group that the ASA 1000V uses for outgoing requests.</td>
</tr>
<tr>
<td><code>dns server-group DefaultDNS</code></td>
<td>Other DNS server groups can be configured for VPN tunnel groups. See the <code>tunnel-group</code> command in the command reference for more information.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><code>hostname(config)# dns server-group DefaultDNS</code></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>Specifies one or more DNS servers. You can enter all six IP addresses in the same command, separated by spaces, or you can enter each command separately. The ASA 1000V tries each DNS server in order until it receives a response.</td>
</tr>
<tr>
<td><code>name-server ip_address [ip_address2] [ip_address3] [ip_address6]</code></td>
<td><code>hostname(config-dns-server-group)# name-server 10.1.1.5 192.168.1.67 209.165.201.6</code></td>
</tr>
</tbody>
</table>
CHAPTER 6

Configuring DHCP

This chapter describes how to configure the DHCP server and includes the following sections:

- Information About DHCP, page 6-1
- Guidelines and Limitations, page 6-1
- Configuring a DHCP Server, page 6-2
- Configuring DHCP Relay Services, page 6-5
- DHCP Monitoring Commands, page 6-6

Information About DHCP

DHCP provides network configuration parameters, such as IP addresses, to DHCP clients. The ASA 1000V can provide a DHCP server or DHCP relay services to DHCP clients attached to ASA 1000V 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.

Guidelines and Limitations

This section lists the guidelines and limitations for this feature.

- You can configure a DHCP server on each interface of the ASA 1000V. 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 ASA 1000V 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 ASA 1000V 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 ASA 1000V.

## Configuring a DHCP Server

This section describes how to configure a DHCP server provided by the ASA 1000V and includes the following topics:

- Enabling the DHCP Server, page 6-2
- Configuring DHCP Options, page 6-3
- DHCP Monitoring Commands, page 6-6

### Configuring a DHCP Server

The ASA 1000V 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 ASA 1000V DHCP server does not support BOOTP requests.

To enable the DHCP server on an ASA 1000V interface, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> <code>dhcpd address ip_address-ip_address</code> <code>interface_name</code></td>
<td>Create a DHCP address pool. The ASA 1000V 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 ASA 1000V interface. The <code>interface_name</code> argument specifies the Ethernet interface name.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><code>hostname(config)# dhcpd address 10.0.1.101-10.0.1.110 inside</code></td>
</tr>
<tr>
<td><strong>Step 2</strong> <code>dhcpd dns dns1 [dns2]</code></td>
<td>(Optional) Specifies the IP address(es) of the DNS server(s).</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><code>hostname(config)# dhcpd dns 209.165.201.2 209.165.202.129</code></td>
</tr>
<tr>
<td><strong>Step 3</strong> <code>dhcpd wins wins1 [wins2]</code></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><strong>Example:</strong></td>
<td><code>hostname(config)# dhcpd wins 209.165.201.5</code></td>
</tr>
<tr>
<td><strong>Step 4</strong> <code>dhcpd lease lease_length</code></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><strong>Example:</strong></td>
<td><code>hostname(config)# dhcpd lease 3000</code></td>
</tr>
</tbody>
</table>
### Configuring DHCP Options

You can configure the ASA 1000V 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 6-3**
- **Options that Return a Text String, page 6-3**
- **Options that Return a Hexadecimal Value, page 6-4**

The ASA 1000V 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>
<tbody>
<tr>
<td><code>dhcpd option code ip addr_1 [addr_2]</code></td>
<td>Configures a DHCP option that returns one or two IP addresses.</td>
</tr>
</tbody>
</table>

**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>
<tbody>
<tr>
<td><code>dhcpd option code ascii text</code></td>
<td>Configures a DHCP option that returns a text string.</td>
</tr>
</tbody>
</table>

**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:

hostname(config)# dhcpd option 2 hex
22.0011.01.FF1111.00FF.0000.AAAA.1111.1111.1111.11

You can configure the ASA 1000V 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:

The ASA 1000V 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 ASA 1000V 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 6-1 shows the DHCP options that the dhcpd option command does not support.

Table 6-1 Unsupported DHCP Options

<table>
<thead>
<tr>
<th>Option Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>DHCP_OPT_PAD</td>
</tr>
<tr>
<td>1</td>
<td>DHCP_OPT_SUBNET_MASK</td>
</tr>
<tr>
<td>12</td>
<td>DHCP_OPT_HOST_NAME</td>
</tr>
<tr>
<td>50</td>
<td>DHCP_OPT_REQUESTED_ADDRESS</td>
</tr>
<tr>
<td>51</td>
<td>DHCP_OPT_LEASE_TIME</td>
</tr>
<tr>
<td>52</td>
<td>DHCP_OPT_OPTION_OVERLOAD</td>
</tr>
<tr>
<td>53</td>
<td>DHCP_OPT_MESSAGE_TYPE</td>
</tr>
<tr>
<td>54</td>
<td>DHCP_OPT_SERVER_IDENTIFIER</td>
</tr>
<tr>
<td>58</td>
<td>DHCP_OPT_RENEWAL_TIME</td>
</tr>
<tr>
<td>59</td>
<td>DHCP_OPT_REBINDING_TIME</td>
</tr>
<tr>
<td>61</td>
<td>DHCP_OPT_CLIENT_IDENTIFIER</td>
</tr>
<tr>
<td>67</td>
<td>DHCP_OPT_BOOT_FILE_NAME</td>
</tr>
<tr>
<td>82</td>
<td>DHCP_OPT_RELAY_INFORMATION</td>
</tr>
<tr>
<td>255</td>
<td>DHCP_OPT_END</td>
</tr>
</tbody>
</table>
### Configuring DHCP Relay Services

A DHCP relay agent allows the ASA 1000V 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 ASA 1000V and cannot send requests through another relay agent or a router.
- When DHCP relay is enabled and more than one DHCP relay server is defined, the ASA 1000V 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 ASA 1000V receives any of the following DHCP messages: ACK, NACK, or decline.

<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 argument is the IP address or name of the primary TFTP server while the server_ip2 argument 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
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>dhcpd option 3 ip router_ip1</td>
<td>Sets the default route.</td>
</tr>
</tbody>
</table>

**Example:**
```
hostname(config)# dhcpd option 3 ip 10.10.1.1
```
You cannot enable DHCP relay on an interface running DHCP proxy. You must remove the VPN DHCP configuration first, or an error message appears. This error occurs if both DHCP relay and DHCP proxy are enabled. Make sure that either DHCP relay or DHCP proxy is enabled, but not both.

To enable DHCP relay, perform the following steps:

**DHCP Monitoring Commands**

To monitor DHCP, enter one or more 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>
Configuring Static and Default Routes

This chapter describes how to configure static and default routes on the ASA 1000V and includes the following sections:

- Information About Routing, page 7-1
- Configuring Static and Default Routes, page 7-4
- Disabling Proxy ARPs, page 7-7
- Monitoring a Static or Default Route, page 7-8
- Configuration Examples for Static or Default Routes, page 7-10

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.

The ASA 1000V only supports static routing.

This section includes the following topics:

- Switching, page 7-1
- How Routing Behaves Within the ASA 1000V, page 7-2

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.
Information About Routing

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.

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 ASA 1000V is not directly connected; for example, when there is a router between a network and the ASA 1000V.

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
```

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 ASA 1000V.

How Routing Behaves Within the ASA 1000V

The ASA 1000V uses both the routing table and XLATE tables for routing decisions. To handle destination IP translated traffic, that is, untranslated traffic, the ASA 1000V searches for existing XLATE tables, or static translation to select the egress interface.

This section includes the following topics:

- Egress Interface Selection Process, page 7-2
- Next Hop Selection Process, page 7-3

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 ASA 1000V 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 the XLATE is created. Incoming return packets are forwarded using an 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 ASA 1000V is possible only for multiple next hops available using a single egress interface. Load sharing cannot share multiple egress interfaces.

When there are no route flaps on the ASA 1000V itself, but some routing process is flapping around it, then destination translated traffic is still forwarded using the old XLATE, not via the route table, until the 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.

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 ASA 1000V and around it. That is, ensure that destination-translated packets that belong to the same flow are always forwarded the same way through the ASA 1000V.

Information About the Routing Table

This section includes the following topics:

- Displaying the Routing Table, page 7-3
- How Forwarding Decisions Are Made, page 7-4

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
```
How Forwarding Decisions Are Made

Forwarding decisions are made according to the following guidelines:

- 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 Ethernet interface of an ASA 1000V 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 192.168.32.0/24 has the longest prefix within the routing table (24 bits compared to 19 bits). Longer prefixes are always preferred over shorter ones when forwarding a packet.

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 Default Static Route, page 7-4
- Configuring a Static Route, page 7-6

Configuring a Default Static Route

A default route identifies the gateway IP address to which the ASA 1000V 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**
If you have two default routes configured on different interfaces that have different metrics, the connection to the ASA 1000V that is made from the higher metric interface fails, but connections to the ASA 1000V 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 try 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 ASA 1000V 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.

**Restrictions**

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 routing for tunneled traffic is not supported.
Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>route if_name 0.0.0.0 0.0.0.0</td>
<td>Enables you to add a static route.</td>
</tr>
<tr>
<td>[distance</td>
<td>tunneled]</td>
</tr>
<tr>
<td>Example:</td>
<td>The dest_ip and mask arguments indicate the IP address for the destination network and the gateway_ip 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 ASA 1000V and performing NAT. The default administrative distance for static routes is 1, giving it precedence over routes discovered by dynamic routing protocols but not directly connect routes. Connected routes always take precedence over static or dynamically discovered routes. 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:</td>
</tr>
<tr>
<td>hostname(config)# route outside 0 0 192.168.2.4 tunneled</td>
<td>Note: You can only create a route for the management interface in the VNMC mode. You can create a route for the management interface and all Ethernet interfaces in the ASDM mode. You must create a route at the management interface using the route command when configuring routing in both the VNMC and ASDM modes.</td>
</tr>
</tbody>
</table>

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. As a result, 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 ASA 1000V, 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 7-7

### 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.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td><code>hostname(config)# route outside 10.10.10.0 255.255.255.0 192.168.1.1 [1]</code></td>
<td>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 ASA 1000V 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>

**Examples**

The following example shows static routes that are equal cost routes that direct traffic to three different gateways on the outside interface. The ASA 1000V distributes the traffic among the specified gateways.

```plaintext
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
```

### 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 ASA 1000V uses proxy ARP when you configure NAT and specify a mapped address that is on the same network as the ASA 1000V Ethernet interface. The only way traffic can reach the hosts is if the ASA 1000V 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 an IPsec site-to-site client address pool that overlaps with an existing network, the ASA 1000V 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 IPsec site-to-site 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 them.

To disable proxy ARPs, enter the following command:

```
sysopt noproxyarp interface
```

### Command Purpose

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>sysopt noproxyarp</code></td>
<td>Disables proxy ARPs.</td>
</tr>
</tbody>
</table>

**Example:**
```
hostname(config)# sysopt noproxyarp exampleinterface
```

---

## 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 Ethernet interface on the ASA 1000V 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 ASA 1000V 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 ASA 1000V needs to communicate with
- A persistent network object on the destination network
Note

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.

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 icmpEcho 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 icmpEcho 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
**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 ASA 1000V 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 ASA 1000V 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 ASA 1000V 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
```
PART 3

Configuring Objects and Access Lists
Configuring Objects

Objects are reusable components for use in your configuration. They can be defined and used in ASA 1000V 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 8-1
- Configuring Regular Expressions, page 8-10
- Scheduling Extended Access List Activation, page 8-14

Configuring Objects and Groups

This section includes the following topics:

- Information About Objects and Groups, page 8-1
- Guidelines and Limitations for Objects and Groups, page 8-2
- Configuring Objects, page 8-3
- Configuring Object Groups, page 8-5
- Monitoring Objects and Groups, page 8-10
- Configuring Regular Expressions, page 8-10

Information About Objects and Groups

The ASA 1000V 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 8-2
Information About Objects

Objects are created in and used by the ASA 1000V 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.

Guidelines and Limitations for Objects and Groups

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.
- Unique qualified names for object groups are supported only in the VNMC mode on the ASA 1000V.
Configuring Objects

This section includes the following topics:

- Configuring a Network Object, page 8-3
- Configuring a Service Object, page 8-3

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 12, “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>object service obj_name</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname(config)# object-service SERVOBJECT1</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>service (protocol</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname(config-service-object)# service tcp source eq www destination eq ssh</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 8-5
- Adding a Network Object Group, page 8-6
- Adding a Service Object Group, page 8-7
- Adding an ICMP Type Object Group, page 8-8
- Nesting Object Groups, page 8-8
- Removing Object Groups, page 8-10

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>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>object-group protocol obj_grp_id&lt;br&gt;Example: hostname(config)# object-group protocol tcp_udp_icmp</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>description text&lt;br&gt;Example: hostname(config-protocol)# description New Group</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>protocol-object protocol&lt;br&gt;Example: hostname(config-protocol)# protocol-object tcp</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

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>object-group network grp_id</td>
<td>Adds a network group.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname (config)# object-group network</td>
<td></td>
</tr>
<tr>
<td>admins</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Adds a network group.</td>
</tr>
<tr>
<td></td>
<td>The grp_id is a text string up to 64</td>
</tr>
<tr>
<td></td>
<td>characters in length and can be any</td>
</tr>
<tr>
<td></td>
<td>combination of letters, digits, and</td>
</tr>
<tr>
<td></td>
<td>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 protocol configuration</td>
</tr>
<tr>
<td></td>
<td>mode.</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td>description text</td>
<td>(Optional) Adds a description. The description</td>
</tr>
<tr>
<td>Example:</td>
<td>can be up to 200 characters.</td>
</tr>
<tr>
<td>hostname (config-network)# Administrator</td>
<td></td>
</tr>
<tr>
<td>Addresses</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:</td>
<td>The object keyword adds an additional object</td>
</tr>
<tr>
<td>hostname (config-network)# network-object</td>
<td>to the network object group.</td>
</tr>
<tr>
<td>host 10.2.2.4</td>
<td>Defines the networks in the group. Enter</td>
</tr>
<tr>
<td></td>
<td>the command for each network or address.</td>
</tr>
</tbody>
</table>

**Example**

To create a network group that includes the IP addresses of three administrators, enter the following commands:

```plaintext
hostname (config)# object-group network admins
hostname (config)# 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></td>
</tr>
</tbody>
</table>
| `object-group service grp_id {tcp | udp | tcp-udp}` | Adds a service group.
| Example: | 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 (port53).
| | The prompt changes to service configuration mode. |

| Step 2 | description text | (Optional) Adds a description. The description can be up to 200 characters. |
| Example: | hostname(config-service)# description DNS Group |

| 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-5. |
| Example: | hostname(config-service)# port-object eq domain |

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
hostname (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
```
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>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><code>object-group icmp-type grp_id</code></td>
</tr>
<tr>
<td><em>Example</em>:</td>
<td><code>hostname(config)# object-group icmp-type ping</code></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><code>description text</code></td>
</tr>
<tr>
<td><em>Example</em>:</td>
<td><code>hostname(config-icmp-type)# description Ping Group</code></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td><code>icmp-object icmp-type</code></td>
</tr>
<tr>
<td><em>Example</em>:</td>
<td><code>hostname(config-icmp-type)# icmp-object echo-reply</code></td>
</tr>
</tbody>
</table>

**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)# 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.
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 8-5), and then perform the steps in this section.

**Detailed Steps**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** object-group group (protocol | network | icmp-type) grp_id | service grp_id (tcp | udp | tcp-udp)) | Adds or edits the specified object group type under which you want to nest another object group. The service_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 “.” |
| Example: hostname(config)# object-group network Engineering_group |                                                                                           |
| **Step 2** group-object group_id | Adds the specified group under the object group you specified in Step 1. The nested group must be of the same type. You can mix and match nested group objects and regular objects within an object group. |
| 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:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>no object-group &lt;grp_id&gt;</td>
<td>Removes the specified object group. The &lt;grp_id&gt; 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>underscore “_”</td>
<td></td>
</tr>
<tr>
<td>dash “-”</td>
<td></td>
</tr>
<tr>
<td>period “.”</td>
<td></td>
</tr>
</tbody>
</table>

**Example:**

```
hostname(config)# no object-group Engineering_host
```

| clear object-group {protocol | network | services | icmp-type} | 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>show access-list</td>
<td>Displays the access list entries that are expanded out into individual entries without their object groupings.</td>
</tr>
</tbody>
</table>

| show running-config object-group | Displays all current object groups. |
| show running-config object-group <grp_id> | Displays the current object groups by their group ID. |
| show running-config object-group <grp_type> | Displays the current object groups by their group type. |

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 8-11
- Creating a Regular Expression Class Map, page 8-13
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.

As an optimization, the ASA 1000V 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 8-1 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, d.g matches dog, dag, dig, and any word that contains those characters, such as doggonnit.</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, dog</td>
</tr>
<tr>
<td>?</td>
<td>Question mark</td>
<td>A quantifier that indicates that there are 0 or 1 of the previous expression. For example, lo?se matches lse or lose.</td>
</tr>
<tr>
<td></td>
<td>Asterisk</td>
<td>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.</td>
</tr>
<tr>
<td>*</td>
<td>Plus</td>
<td>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.</td>
</tr>
<tr>
<td>{x} or {x,}</td>
<td>Minimum repeat quantifier</td>
<td>Repeat at least x times. For example, ab(xy){2,}z matches abxyxyz, abxyxyxyz, and so on.</td>
</tr>
</tbody>
</table>

Note: You must enter Ctrl+V and then the question mark or else the help function is invoked.
### Table 8-1  regex Metacharacters (continued)

<table>
<thead>
<tr>
<th>Character</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>[abc]</td>
<td>Character class</td>
<td>Matches any character in the brackets. For example, [abc] matches a, b, or c.</td>
</tr>
<tr>
<td>[^abc]</td>
<td>Negated character class</td>
<td>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.</td>
</tr>
<tr>
<td>[a-c]</td>
<td>Character range class</td>
<td>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].</td>
</tr>
<tr>
<td>&quot;&quot;</td>
<td>Quotation marks</td>
<td>Preserves trailing or leading spaces in the string. For example, “test” preserves the leading space when it looks for a match.</td>
</tr>
<tr>
<td>^</td>
<td>Caret</td>
<td>Specifies the beginning of a line.</td>
</tr>
<tr>
<td>\</td>
<td>Escape character</td>
<td>When used with a metacharacter, matches a literal character. For example, [ matches the left square bracket.</td>
</tr>
<tr>
<td>char</td>
<td>Character</td>
<td>When character is not a metacharacter, matches the literal character.</td>
</tr>
<tr>
<td>\r</td>
<td>Carriage return</td>
<td>Matches a carriage return 0x0d.</td>
</tr>
<tr>
<td>\n</td>
<td>Newline</td>
<td>Matches a new line 0x0a.</td>
</tr>
<tr>
<td>\t</td>
<td>Tab</td>
<td>Matches a tab 0x09.</td>
</tr>
<tr>
<td>\f</td>
<td>Formfeed</td>
<td>Matches a form feed 0x0c.</td>
</tr>
<tr>
<td>\xNN</td>
<td>Escaped hexadecimal number</td>
<td>Matches an ASCII character using hexadecimal (exactly two digits).</td>
</tr>
<tr>
<td>\NNN</td>
<td>Escaped octal number</td>
<td>Matches an ASCII character as octal (exactly three digits). For example, the character 040 represents a space.</td>
</tr>
</tbody>
</table>

### 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
hostname(config)# regex url_example2 example2\.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

hostname(config-cmap)#

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\.[a-z]{1,}
hostname(config)# regex url_example2 example2\.[a-z]{1,}
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 8-14
- Guidelines and Limitations for Scheduling Access List Activation, page 8-14
- Guidelines and Limitations for Scheduling Access List Activation, page 8-14
- Configuring and Applying Time Ranges, page 8-15
- Configuration Examples for Scheduling Access List Activation, page 8-16

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.

Guidelines and Limitations for Scheduling Access List Activation

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 ASA 1000V 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>time-range name</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>hostname(config)# time range Sales</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 2</th>
<th>Do one of the following:</th>
</tr>
</thead>
<tbody>
<tr>
<td>periodic days-of-the-week time</td>
<td>Specifies a recurring time range.</td>
</tr>
<tr>
<td>periodic monday 7:59 to friday 17:01</td>
<td>You can specify the following values for days-of-the-week:</td>
</tr>
<tr>
<td>daily</td>
<td>• monday, tuesday, wednesday, thursday, friday, saturday, or sunday.</td>
</tr>
<tr>
<td>weekdays</td>
<td>• daily</td>
</tr>
<tr>
<td>weekend</td>
<td>• weekdays</td>
</tr>
<tr>
<td>absolute start time date</td>
<td>• weekend</td>
</tr>
<tr>
<td>absolute start 7:59 2 january 2009</td>
<td>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.</td>
</tr>
</tbody>
</table>

| **Step 3** | access-list access_list_name [extended] \(\{\text{deny} \mid \text{permit}\}...[\text{time-range name}]\) | Applies the time range to an ACE. |
| **Example:** | hostname(config)# access list Marketing extended deny tcp host 209.165.200.225 host 209.165 201.1 time-range Pacific_Coast | |

**Note** If you also enable logging for the ACE, use the log keyword before the time-range keyword. If you disable the ACE using the inactive keyword, use the inactive keyword as the last keyword.

See Chapter 9, “Adding an Extended Access List,” for complete access-list command syntax.

**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
```
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 Access Lists, page 9-1
- Guidelines and Limitations, page 9-3
- Default Settings, page 9-3
- Configuring Extended Access Lists, page 9-3
- Monitoring Extended Access Lists, page 9-6
- Configuration Examples for Extended Access Lists, page 9-6
- Where to Go Next, page 9-8

Information About Access Lists

Cisco ASA 1000Vs 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 14, “Configuring a Service Policy Using the Modular Policy Framework.”

This chapter includes the following sections:

- Access Control Entry Order, page 9-2
- Access Control Implicit Deny, page 9-2
- IP Addresses Used for Access Lists When You Use NAT, page 9-2
**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 ASA 1000V decides whether to forward or to drop a packet, the ASA 1000V 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 (except Extended 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 ASA 1000V 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

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
Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

- 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-5.

- 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-5. 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 ASA 1000V 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 9-1 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>log</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 9-4
- Adding Remarks to Access Lists, page 9-6
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 8-1.

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>
</table>
| **(For IP traffic, no ports)** access-list access_list_name [line line_number] extended (deny | Adds an extended ACE. The line line_number option specifies the line number at which insert the ACE. If you do not specify a line number, the ACE is added to the end of the access list. The line number is not saved in the configuration; it only specifies where to insert the ACE. The deny keyword denies a packet if the conditions are matched. The permit keyword permits a packet if the conditions are matched. Instead of entering the protocol, IP address, or port directly in the command, you can use network objects, or protocol, network, port, or ICMP object groups using the object and object-group keyword. See “Configuring Objects and Groups” section on page 8-1 for more information about creating objects. Instead of entering the protocol, IP address, or port directly in the command, you can use network objects, or protocol, network, port, or ICMP object groups using the object and object-group keyword. See “Configuring Objects and Groups” section on page 8-1 for more information about creating objects. The protocol argument specifies the IP protocol name or number. For example UDP is 17, TCP is 6, and EGP is 47. The source_address specifies the IP address of the network or host from which the packet is being sent. Enter the host keyword before the IP address to specify a single address. In this case, do not enter a mask. Enter the any keyword instead of the address and mask to specify any address. For the TCP and UDP protocols only, the operator port option matches the port numbers used by the source or destination. The permitted operators are as follows:  - lt—less than.  - gt—greater than.  - eq—equal to.  - neq—not equal to.  - range—an inclusive range of values. When you use this operator, specify two port numbers, for example: range 100 200. The dest_address argument specifies the IP address of the network or host to which the packet is being sent. Enter the host keyword before the IP address to specify a single address. In this case, do not enter a mask. Enter the any keyword instead of the address and mask to specify any address. The icmp_type argument specifies the ICMP type if the protocol is ICMP. The time-range keyword specifies when an access list is activated. See the “Scheduling Extended Access List Activation” section on page 8-14 for more information. The inactive keyword disables an ACE. To reenable it, enter the entire ACE without the inactive keyword. This feature enables you to keep a record of an inactive ACE in your configuration to make reenabling easier. For the log keyword, see Chapter 10, “Configuring Logging for Access Lists.”

Example:
hostname(config)# access-list ACL_IN extended permit ip any any

(For TCP or UDP traffic, with ports)
access-list access_list_name [line line_number] extended (deny | permit) tcp | udp | object-group prot_grp_id) (source_address mask | object nw_obj_id | object-group nw_grp_id) [operator port | object-group svc_grp_id] (dest_address mask | object nw_obj_id | object-group nw_grp_id) [log [[level] [interval secs] | disable | default] [inactive | time-range time_range_name]

(For ICMP traffic)
access-list access_list_name [line line_number] extended (deny | permit) icmp (source_address mask | object nw_obj_id | object-group nw_grp_id) (dest_address mask | object nw_obj_id | object-group nw_grp_id) [icmp_type | object-group icmp_grp_id] [log [[level] [interval secs] | disable | default] [inactive | time-range time_range_name]
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 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:

```
show access list
show running-config access-list
```

Configuration Examples for Extended Access Lists

This section includes the following topics:

- Configuration Examples for Extended Access Lists (No Objects), page 9-7
- Configuration Examples for Extended Access Lists (Using Objects), page 9-7
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 ASA 1000Ve:

```
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-network)# network-object host 10.1.1.4
hostname(config-network)# network-object host 10.1.1.78
hostname(config-network)# network-object host 10.1.1.89

hostname(config-network)# object-group network web
hostname(config-network)# network-object host 209.165.201.29
hostname(config-network)# network-object host 209.165.201.16
hostname(config-network)# network-object host 209.165.201.78

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

---

Where to Go Next

Apply the access list to an interface. See the “Configuring Access Rules” section on page 16-4 for more information.
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 10-1
- Managing Deny Flows, page 10-4

Configuring Logging for Access Lists

This section includes the following topics:

- Information About Logging Access List Activity, page 10-1
- Guidelines and Limitations, page 10-2
- Default Settings, page 10-2
- Configuring Access List Logging, page 10-3
- Monitoring Access Lists, page 10-3
- Configuration Examples for Access List Logging, page 10-4
- Managing Deny Flows, page 10-4

Information About Logging Access List Activity

By default, when traffic is denied by an extended ACE or a Webtype ACE, the ASA 1000V generates syslog message 106023 for each denied packet in the following form:

%ASA-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 ASA 1000V 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.
Note
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-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 ASA 1000V creates a flow entry to track the number of packets received within a specific interval. The ASA 1000V 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 ASA 1000V resets the hit count to 0. If no packets match the ACE during an interval, the ASA 1000V 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 10-4 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` for detailed information about this syslog message.

## 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 10-1 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 <code>log</code> 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 section describes how to configure access list logging.

Note: For complete access list command syntax, see the “Configuring Extended Access Lists” section on page 9-3.

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 1.1.1.1 any log 7 interval 600
```

**Command Purpose**

- **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.
  - 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:

```
show access list
show running-config access-list
```

**Command Purpose**

- **show access list**
  - Displays the access list entries by number.

- **show running-config access-list**
  - Displays the current running access list configuration.
Chapter 10      Configuring Logging for Access Lists

Managing Deny Flows

This section includes the following topics:

- Information About Managing Deny Flows, page 10-4
- Licensing Requirements for Managing Deny Flows, page 10-5
- Guidelines and Limitations, page 10-5
- Managing Deny Flows, page 10-6
- Monitoring Deny Flows, page 10-6

Information About Managing Deny Flows

When you enable logging for message 106100, if a packet matches an ACE, the ASA 1000V creates a flow entry to track the number of packets received within a specific interval. The ASA 1000V 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 ASA 1000V 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 ASA 1000V does not create a new deny flow for logging until the existing flows expire.

For example, if someone initiates a DoS attack, the ASA 1000V 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 ASA 1000V issues syslog message 106100: 

```
%ASA-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 ASA 1000V 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 context mode.

**Firewall Mode Guidelines**

Supported only in routed firewall modes

**Additional Guidelines and Limitations**

The ASA 1000V places a limit on the number of concurrent *deny* flows only—not permit flows.

### Default Settings

Table 10-1 lists the default settings for managing deny flows.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>numbers</code></td>
<td>The <code>numbers</code> argument specifies the maximum number of deny flows. The default is 4096.</td>
</tr>
<tr>
<td><code>secs</code></td>
<td>The <code>secs</code> 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 <code>number</code> 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 <code>secs</code> 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>
PART 4

Configuring Network Address Translation
Information About NAT

This chapter provides an overview of how Network Address Translation (NAT) works on the ASA 1000V. This chapter includes the following sections:

- Why Use NAT?, page 11-1
- NAT Terminology, page 11-3
- NAT Types, page 11-3
- NAT for VPN, page 11-13
- NAT for VPN, page 11-13
- How NAT is Implemented, page 11-14
- NAT Rule Order, page 11-18
- Routing NAT Packets, page 11-20
- DNS and NAT, page 11-22
- Where to Go Next, page 11-24

Note

To start configuring NAT, see Chapter 12, “Configuring Network Object NAT,” or Chapter 13, “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
Why Use NAT?

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.

Figure 11-1 shows a typical NAT example, with a private network on the inside.

Figure 11-1  NAT Example

1. When the HR 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 ASA 1000V receives the packet because the ASA 1000V performs proxy ARP to claim the packet.
3. The ASA 1000V 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 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 ASA 1000V, 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 11-3
- **Static NAT**, page 11-4
- **Dynamic NAT**, page 11-9
- **Dynamic PAT**, page 11-11
- **Identity NAT**, page 11-12

**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 11-4.

- **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 11-9.

- **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 11-11.

- **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 11-12.
Static NAT

This section describes static NAT and includes the following topics:

- Information About Static NAT, page 11-4
- Information About Static NAT with Port Translation, page 11-4
- Information About One-to-Many Static NAT, page 11-7
- Information About Other Mapping Scenarios (Not Recommended), page 11-8

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 11-2 shows a typical static NAT scenario. The translation is always active so both real and remote hosts can initiate connections.

![Figure 11-2 Static NAT](image)

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 11-4
- Static NAT with Identity Port Translation, page 11-6
- Static NAT with Port Translation for Non-Standard Ports, page 11-6
- Static Interface NAT with Port Translation, page 11-6

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 11-3 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 ASA 1000V 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 11-4. See the “Single Address for FTP, HTTP, and SMTP (Static NAT-with-Port-Translation)” section on page 12-17 for details on how to configure this example.)

**Figure 11-4  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 ASA 1000V security profile interface to an outside host, then you can map the outside host IP address/port 23 to the ASA 1000V interface address/port 23. Because the inside security profile interfaces do not have IP addresses, you cannot use static interface PAT on the inside.
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 11-5 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 11-5 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 11-6). (See the “Inside Load Balancer with Multiple Mapped Addresses (Static NAT, One-to-Many)” section on page 12-16 for details on how to configure this example.)

Figure 11-6 One-to-Many Static NAT

Information About Other Mapping Scenarios (Not Recommended)

The ASA 1000V 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 11-7 shows a typical few-to-many static NAT scenario.

![Few-to-Many Static NAT Diagram]

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 11-8 shows a typical many-to-few static NAT scenario.

![Many-to-Few Static NAT Diagram]

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 11-10
- Dynamic NAT Disadvantages and Advantages, page 11-11
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 ASA 1000V 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 11-9 shows a typical dynamic NAT scenario. Only real hosts can create a NAT session, and responding traffic is allowed back.

**Figure 11-9** Dynamic NAT

**Figure 11-10** shows a remote host attempting to initiate a connection to a mapped address. This address is not currently in the translation table; therefore, the ASA 1000V drops the packet.

**Figure 11-10** 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 19-3 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 11-11
- Dynamic PAT Disadvantages and Advantages, page 11-12

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. (8.4(3) and later, not including 8.5(1)) If you have a lot of traffic that uses the lower port ranges, you can now specify a flat range of ports to be used instead of the three unequal-sized tiers.

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 11-11 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.

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).

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 ASA 1000V outside interface IP address as the PAT address. (The inside security profile interfaces do not have IP addresses, so you cannot use interface PAT on the inside).

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 19-3 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. 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 11-12 shows a typical identity NAT scenario.

**Figure 11-12  Identity NAT**

NAT for VPN

Figure 11-13 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.

**Figure 11-13  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
3. IM received

A. HTTP to www.example.com
B. ASA performs interface PAT for outgoing traffic.
C. HTTP request to www.example.com
How NAT is Implemented

The ASA 1000V 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 11-14
- Information About Network Object NAT, page 11-15
- Information About Twice NAT, page 11-15

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 11-18 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 ASA 1000V, 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 12, “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 13, “Configuring Twice NAT.”

Figure 11-14 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 12-17 for details on how to configure this example.)
How NAT is Implemented

Figure 11-14  Twice NAT with Different Destination Addresses

Figure 11-15 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 11-15  Twice NAT with Different Destination Ports
Figure 11-16 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 11-16 Twice Static NAT with Destination Address Translation
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 11-1 shows the order of rules within each section.

<table>
<thead>
<tr>
<th>Section</th>
<th>Rule Type</th>
<th>Order of Rules within the Section</th>
</tr>
</thead>
</table>
| Section 1 | Twice NAT         | Applied on a first match basis, in the order they appear in the configuration. By default, twice NAT rules are added to section 1.  
 Note: If you configure EasyVPN remote, the ASA 1000V 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. |
| Section 2 | Network object NAT | Section 2 rules are applied in the following order, as automatically determined by the ASA 1000V:  
1. Static rules.  
2. Dynamic rules.  
Within each rule type, the following ordering guidelines are used:  
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.  
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.  
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. |
| Section 3 | Twice NAT         | 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. |

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**

NAT rules applied to the outside interface must refer to the outside Ethernet interface directly. For security policy purposes, the inside interface is divided up into separate security profiles. NAT rules applied to the inside interface must refer to a specific security profile, and not the inside interface directly.

**Note**

Because security profile interfaces do not have IP addresses, you cannot use the security profile interface for interface PAT.

You can configure NAT between security profiles and the outside, but you cannot configure NAT between security profiles; because hosts defined by security profiles are all on the inside interface, traffic between security profiles does not pass through the ASA 1000V; they can reach each other directly or through the VSG if desired.

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 11-17).

**Figure 11-17** Specifying Any Interface

### Routing NAT Packets

The ASA 1000V 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 ASA 1000V handles accepting and delivering packets with NAT, and includes the following topics:

- Mapped Addresses and Routing, page 11-20
- Determining the Egress Interface, page 11-22

### 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 12, “Configuring Network Object NAT,” and Chapter 13, “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 ASA 1000V 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 ASA 1000V 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.
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 ASA 1000V. Alternatively, you can configure a static route on the ASA 1000V for the mapped addresses, and then redistribute the route using your routing protocol.

- **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 ASA 1000V will then proxy ARP for the address, even though the packet is not actually destined for the ASA 1000V. (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 ASA 1000V ARP response is received before the actual host ARP response, then traffic will be mistakenly sent to the ASA 1000V (see Figure 11-18).

**Figure 11-18 Proxy ARP Problems with Identity NAT**

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 ASA 1000V using a service like Telnet before any other traffic can pass. You can configure a virtual Telnet server on the ASA 1000V...
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 ASA 1000V keep traffic destined for the virtual Telnet address rather than send the traffic out the source interface according to the NAT rule. (See Figure 11-19).

**Figure 11-19  Proxy ARP and Virtual Telnet**

---

**Determining the Egress Interface**

The ASA 1000V determines the egress interface for a NAT packet in the following way:

- If you specify an optional interface, then the ASA 1000V 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 ASA 1000V uses a route lookup to determine the egress interface.

**DNS and NAT**

You might need to configure the ASA 1000V 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 ASA 1000V 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 11-20 shows a DNS server that is accessible from the outside interface. A server, ftp.cisco.com, is on the security profile QA interface. You configure the ASA 1000V 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 QA 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 a QA 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 ASA 1000V refers to the static rule for the QA server and translates the address inside the DNS reply to 10.1.3.14. If you do not enable DNS reply modification, then the QA host attempts to send traffic to 209.165.201.10 instead of accessing ftp.cisco.com directly.

Figure 11-20  DNS Reply Modification, DNS Server on Outside
**Figure 11-21** shows a web server and DNS server on the outside. The ASA 1000V has a static translation for the outside server. In this case, when a security profile QA 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 QA 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 11-21    DNS Reply Modification, DNS Server on Host Network**

Where to Go Next

To configure network object NAT, see Chapter 12, “Configuring Network Object NAT.”

To configure twice NAT, see Chapter 13, “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 12-1
- Prerequisites for Network Object NAT, page 12-2
- Guidelines and Limitations, page 12-2
- Default Settings, page 12-2
- Configuring Network Object NAT, page 12-3
- Monitoring Network Object NAT, page 12-13
- Configuration Examples for Network Object NAT, page 12-13
- Feature History for Network Object NAT, page 12-21

Note For detailed information about how NAT works, see Chapter 11, “Information About NAT.”

Information About Network Object NAT

When a packet enters the ASA 1000V, 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 11-14.

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 11-18.
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 discontinous IP address ranges or multiple hosts or subnets. To create a network object or group, see the “Configuring Objects and Groups” section on page 8-1.

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

- 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-10.10.10.1-01, object network obj-10.10.10.1-02, and so on.
- 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, use the interface keyword instead of the IP address.
  - (Dynamic NAT) The standby interface IP address when VPN is enabled.
- You cannot configure interface PAT on the inside security profile interfaces, because they do not have IP addresses.
- For application inspection limitations with NAT or PAT, see the “Default Settings” section on page 19-3 in Chapter 19, “Getting Started with Application Layer Protocol Inspection.”

Default Settings

- 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 11-20 for more information.
• If you specify an optional interface, then the ASA 1000V 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 11-20 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 12-3
• Configuring Dynamic PAT (Hide), page 12-5
• Configuring Static NAT or Static NAT-with-Port-Translation, page 12-9
• Configuring Identity NAT, page 12-11

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 11-9.

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 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.</td>
</tr>
<tr>
<td>object network obj_name</td>
<td></td>
</tr>
<tr>
<td>range ip_address_1 ip_address_2</td>
<td></td>
</tr>
<tr>
<td>Network object group:</td>
<td></td>
</tr>
<tr>
<td>object-group network grp_name</td>
<td></td>
</tr>
<tr>
<td>(network-object {object net_obj_name</td>
<td></td>
</tr>
<tr>
<td>host ip_address}</td>
<td></td>
</tr>
<tr>
<td>group-object grp_obj_name)</td>
<td></td>
</tr>
</tbody>
</table>

Example:

hostname(config)# object network TEST
hostname(config-network-object)# range 10.1.1.1 10.1.1.70
hostname(config)# object network TEST2
hostname(config-network-object)# range 10.1.2.1 10.1.2.70
hostname(config-network-object)#
object-group network MAPPED_IPS
hostname(config-network-object)# network-object
object TEST
hostname(config-network-object)# network-object
object TEST2
hostname(config-network-object)# network-object host 10.1.2.79
### Configuring Network Object NAT

#### Step 2
**object network** *obj_name*

**Example:**
```
hostname(config)# object network
my-host-obj
```

- **Command:** 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)# subnet 10.1.1.0 255.255.255.0
```

- **Purpose:** If you are creating a new network object, defines the real IP address(es) that you want to translate.

#### Step 4
```
nat ([real_ifc,mapped_ifc]) dynamic
mapped_obj [interface] [dns]
```

**Example:**
```
hostname(config-network-object)# nat (VM1,outside) dynamic MAPPED_IPS interface
```

- **Purpose:** Configures **dynamic NAT** for the object IP addresses.

**Note:** You can only define a single NAT rule for a given object. See the “Guidelines and Limitations” section on page 12-2.

See the following guidelines:

- **Interfaces**—Specify the real and mapped interfaces. Be sure to include the parentheses in your command. 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 the outside interface for the *mapped_ifc*; inside security profile interfaces do not support interface PAT.
- **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 11-22 for more information.

### 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:

```
hostname(config)# object network my-range-obj
hostname(config-network-object)# range 10.2.2.1 10.2.2.10
```

The following example configures dynamic NAT with dynamic PAT backup. Hosts on security profile VM1 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.
hostname(config)# object network nat-range1
hostname(config-network-object)# range 10.10.10.10 10.10.10.20
hostname(config-network-object)# object network pat-ip1
hostname(config-network-object)# host 10.10.10.21
hostname(config-network-object)# object-group network nat-pat-grp
hostname(config-network-object)# network-object object nat-range1
hostname(config-network-object)# network-object object pat-ip1
hostname(config-network-object)# object network my_net_obj5
hostname(config-network-object)# subnet 10.76.11.0 255.255.255.0
hostname(config-network-object)# nat (VM1, outside) dynamic nat-pat-grp 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 11-11.

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. Therefore, ports below 1024 have only a small PAT pool that can be used. If you have a lot of traffic that uses the lower port ranges, you can now specify a flat range of ports to be used instead of the three unequal-sized tiers: either 1024 to 65535, or 1 to 65535.
- If you use the same PAT pool object in two separate rules, then be sure to specify the same options for each rule. For example, if one rule specifies extended PAT and a flat range, then the other rule must also specify extended PAT and a flat range.

For extended PAT for a PAT pool:

- Many application inspections do not support extended PAT. See the “Default Settings” section on page 19-3 in Chapter 19, “Getting Started with Application Layer Protocol Inspection,” for a complete list of unsupported inspections.
- If you enable extended PAT for a dynamic PAT rule, then you cannot also use an address in the PAT pool as the PAT address in a separate static NAT-with-port-translation rule. For example, if the PAT pool includes 10.1.1.1, then you cannot create a static NAT-with-port-translation rule using 10.1.1.1 as the PAT address.
- If you use a PAT pool and specify an interface for fallback, you cannot specify extended PAT.
- For VoIP deployments that use ICE or TURN, do not use extended PAT. ICE and TURN rely on the PAT binding to be the same for all destinations.

For round robin for a PAT pool:

- If a host has an existing connection, then subsequent connections from that host will use the same PAT IP address if ports are available. Note: This “stickiness” does not survive a failover. If the ASA 1000V fails over, then subsequent connections from a host may not use the initial IP address.
• Round robin, especially when combined with extended PAT, can consume a large amount of memory. Because NAT pools are created for every mapped protocol/IP address/port range, round robin results in a large number of concurrent NAT pools, which use memory. Extended PAT results in an even larger number of concurrent NAT pools.

### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>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 <code>nat</code> command or if you want to use the interface address by specifying the <code>interface</code> keyword. For mapped addresses used as a PAT pool, all addresses in the object or group, including ranges, are used as PAT addresses. <strong>Note</strong> The object or group cannot contain a subnet. See the “Guidelines and Limitations” section on page 12-2 for information about disallowed mapped IP addresses. For more information about configuring a network object or group, see the “Configuring Objects” section on page 8-3.</td>
</tr>
<tr>
<td>(Optional)</td>
<td></td>
</tr>
<tr>
<td>Network object:</td>
<td></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>`{host ip_address</td>
<td>range ip_address_1 ip_address_2}`</td>
</tr>
<tr>
<td>Network object group:</td>
<td></td>
</tr>
<tr>
<td><code>object-group network grp_name</code></td>
<td></td>
</tr>
<tr>
<td>`{object-object {object net_obj_name</td>
<td>host ip_address}</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# object network PAT_POOL1 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 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 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>hostname(config-network-object)# network-object object PAT_POOL1 object PAT_POOL2</td>
<td></td>
</tr>
<tr>
<td>hostname(config-network-object)# network-object object PAT_IP</td>
<td></td>
</tr>
<tr>
<td>Step 2</td>
<td></td>
</tr>
<tr>
<td><code>object network obj_name</code></td>
<td>If you are creating a new network object, defines the real IP address(es) that you want to translate.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# object network my-host-obj1</td>
<td></td>
</tr>
<tr>
<td>Step 3</td>
<td></td>
</tr>
<tr>
<td>`(host ip_address</td>
<td>subnet subnet_address netmask</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>hostname(config-network-object)# range 10.1.1.1 10.1.1.90</td>
<td></td>
</tr>
</tbody>
</table>
Step 4

```
nat [(real_ifc,mapped_ifc)] dynamic
    (mapped_inline_host_ip | mapped_obj | pat-pool mapped_obj [round-robin]
    [extended] [flat [include-reserve]] | interface) [interface] [dns]
```

Example:
```
hostname(config-network-object)# nat
(any,outside) dynamic interface
```

Configures dynamic PAT for the object IP addresses. You can only define a single NAT rule for a given object. See the “Guidelines and Limitations” section on page 12-2.

See the following guidelines:

- Interfaces—Specify the real and mapped interfaces. Be sure to include the parentheses in your command. 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—The IP address of the mapped interface is used as the mapped address. For this option, you must configure the outside interface for the mapped_ifc; inside security profile interfaces do not support interface PAT. 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)
Chapter 12  Configuring Network Object NAT

## Configuring Network Object NAT

### 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 (VM1,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 (VM1,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 11-4.

### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> <em>(Optional)</em>&lt;br&gt;Network object:&lt;br&gt;object network obj_name&lt;br&gt;  (host ip_address</td>
<td>subnet subnet_address netmask</td>
</tr>
<tr>
<td>Network object group:&lt;br&gt;object-group network grp_name&lt;br&gt;  (network-object (object net_obj_name</td>
<td>subnet subnet_address netmask</td>
</tr>
<tr>
<td><strong>Example:</strong>&lt;br&gt;hostname(config)# object network&lt;br&gt;  MAPPED_IPS&lt;br&gt;hostname(config-network-object)# subnet&lt;br&gt;  10.1.1.0 255.255.255.0</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> object network obj_name</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><strong>Example:</strong>&lt;br&gt;hostname(config)# object network&lt;br&gt;  my-host-obj1</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> *(host ip_address</td>
<td>subnet subnet_address netmask</td>
</tr>
<tr>
<td><strong>Example:</strong>&lt;br&gt;hostname(config-network-object)# subnet&lt;br&gt;  10.2.1.0 255.255.255.0</td>
<td></td>
</tr>
</tbody>
</table>
Chapter 12  Configuring Network Object NAT

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]

Example:
hostname(config-network-object)# nat (VM1,outside) static MAPPED_IPS service tcp 80 8080

Command

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>nat ((real_ifc,mapped_ifc)) static (mapped_inline_ip</td>
<td>mapped_obj</td>
</tr>
</tbody>
</table>

Note You can only define a single NAT rule for a given object.
See the “Guidelines and Limitations” section on page 12-2.

See the following guidelines:

- **Interfaces**—Specify the real and mapped interfaces. Be sure to include the parentheses in your command. 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 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) For this option, you must configure the outside interface for the mapped_ifc; inside security profile interfaces do not support interface PAT. 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 11-4.

- **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 11-22 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 11-20 for more information.
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 (VM1,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 (VM1,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 (VM1,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 11-12.

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> <em>(Optional)</em></td>
<td>For the <strong>mapped</strong> 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 <strong>nat</strong> command. For more information about configuring a network object, see the “Configuring Objects” section on page 8-3.</td>
</tr>
<tr>
<td><code>object network obj_name</code> *(host ip_address</td>
<td>subnet subnet_address netmask</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>hostname(config)# object network MAPPED_IPS</code></td>
<td></td>
</tr>
<tr>
<td><code>hostname(config-network-object)# subnet 10.1.1.0 255.255.255.0</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></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><code>object network obj_name</code></td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>hostname(config)# object network my-host-obj1</code></td>
<td></td>
</tr>
</tbody>
</table>
## 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 10.1.1.0 255.255.255.0
```

### Step 4

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>nat ((real_ifc,mapped_ifc)) static {mapped_inline_ip</td>
<td>mapped_obj}</td>
</tr>
<tr>
<td>[no-proxy-arp] [route-lookup]</td>
<td></td>
</tr>
</tbody>
</table>

**Example:**

```
hostname(config-network-object)# nat (VM1,outside) static MAPPED_IPS
```

**Note** You can only define a single NAT rule for a given object. See the “Guidelines and Limitations” section on page 12-2.

See the following guidelines:

- **Interfaces**—Specify the real and mapped interfaces. Be sure to include the parentheses in your command. 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**—Be sure to configure the same IP address for both the mapped and real address. Use one of the following:
  - Network object—Including the same IP address as the real object (see Step 1).
  - 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 10.1.1.1 as the mapped address, then the mapped range will include 10.1.1.1 through 10.1.1.6.

- **No Proxy ARP**—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 11-20 for more information.

- **Route lookup**—(Interface(s) specified) Specify route-lookup to determine the egress interface using a route lookup instead of using the interface specified in the NAT command. See the “Determining the Egress Interface” section on page 11-22 for more information.

### 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 (VM1,outside) static 10.1.1.1
```
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 (VM1, 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 (VM1, outside) dynamic pool
object network network-2
  nat (VM1, outside) dynamic pool
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show xlate</td>
<td>Shows current NAT session information.</td>
</tr>
</tbody>
</table>

## Configuration Examples for Network Object NAT

This section includes the following configuration examples:
Providing Access to an Inside Web Server (Static NAT)

The following example performs static NAT for an inside Eng 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 12-1).

Figure 12-1 Static NAT for an Inside Web Server

Step 1 Create a network object for the Eng 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 (eng,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 servers on a private network when they access the outside. Also, when inside servers 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 12-2).

![Figure 12-2 Dynamic NAT for Inside, Static NAT for Outside Web Server](image)

**Step 1** Create a network object for the dynamic NAT pool to which you want to translate the server 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 Server network:

```
hostname(config)# object network myInsNet
hostname(config-network-object)# subnet 10.1.2.0/24
```

**Step 3** Enable dynamic NAT for the Servers network:

```
hostname(config-network-object)# nat (Servers, outside) dynamic myNatPool
```

**Step 4** Create a network object for the outside web server:

```
hostname(config)# object network outWebServ
```
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,Servers) 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 12-3).

**Figure 12-3  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:

```
hostname(config)# object network myPublicIPs
hostname(config-network-object)# range 209.165.201.3 209.165.201.8
```
**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 (LoadBalance, 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 Finance 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 12-4.)

*Figure 12-4  Static NAT-with-Port-Translation*

**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 identity port translation for the FTP server:
```
UDP 10.1.2.28:209.165.201.3
TCP 10.1.2.28:209.165.201.3
```

**Step 2**  Define the FTP server address, and configure static NAT with identity port translation for the FTP server:
hostname(config-network-object)# host 209.165.201.3
hostname(config-network-object)# nat (outside,Finance) static 10.1.2.28 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 209.165.201.4
hostname(config-network-object)# nat (outside,Finance) static 10.1.2.28 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 209.165.201.5
hostname(config-network-object)# nat (outside,Finance) static 10.1.2.28 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 QA security profile interface. You configure the ASA 1000V 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 12-5.) In this case, you want to enable DNS reply modification on this static rule so that QA 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 a QA 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 ASA 1000V refers to the static rule for the QA server and translates the address inside the DNS reply to 10.1.3.14. If you do not enable DNS reply modification, then the QA host attempts to send traffic to 209.165.201.10 instead of accessing ftp.cisco.com directly.

![Figure 12-5 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 10.1.3.14
hostname(config-network-object)# nat (QA,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 12-6 shows a web server and DNS server on the outside. The ASA 1000V has a static translation for the outside server. In this case, when an inside QA security profile 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 QA 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 12-6 DNS Reply Modification Using Outside NAT**

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,QA) static 10.1.2.56 dns
```
### Feature History for Network Object NAT

Table 12-1 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: <code>nat</code> (object network configuration mode), <code>show nat</code>, <code>show xlate</code>, <code>show nat pool</code>.</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 <code>no-proxy-arp</code> and <code>route-lookup</code> keywords, to maintain existing functionality. We modified the following commands: <code>nat static [no-proxy-arp] [route-lookup]</code>.</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: <code>nat dynamic [pat-pool mapped_object [round-robin]]</code>.</td>
</tr>
<tr>
<td>Round robin PAT pool allocation uses the same IP address for existing hosts</td>
<td>8.4(3)</td>
<td>When using a PAT pool with round robin allocation, if a host has an existing connection, then subsequent connections from that host will use the same PAT IP address if ports are available. We did not modify any commands. This feature is not available in 8.5(1) or 8.6(1).</td>
</tr>
</tbody>
</table>
Feature History for Network Object NAT

Table 12-1 Feature History for Network Object NAT (continued)

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flat range of PAT ports for a PAT pool</td>
<td>8.4(3)</td>
<td>If available, the real source port number is used for the mapped port. However, if the real port is <em>not</em> 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. If you have a lot of traffic that uses the lower port ranges, when using a PAT pool, you can now specify a flat range of ports to be used instead of the three unequal-sized tiers: either 1024 to 65535, or 1 to 65535. We modified the following commands: <code>nat dynamic [pat-pool mapped_object [flat [include-reserve]]].</code></td>
</tr>
</tbody>
</table>

*This feature is not available in 8.5(1) or 8.6(1).*
Chapter 12 Configuring Network Object NAT

Table 12-1 Feature History for Network Object NAT (continued)

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
</table>
| Extended PAT for a PAT pool | 8.4(3) | Each PAT IP address allows up to 65535 ports. If 65535 ports do not provide enough translations, you can now enable extended PAT for a PAT pool. Extended PAT uses 65535 ports per *service*, as opposed to per IP address, by including the destination address and port in the translation information.

We modified the following commands: `nat dynamic [pat-pool mapped_object [extended]]`.

This feature is not available in 8.5(1) or 8.6(1).

Automatic NAT rules to translate a VPN peer’s local IP address back to the peer’s real IP address | 8.4(3) | In rare situations, you might want to use a VPN peer’s real IP address on the inside network instead of an assigned local IP address. Normally with VPN, the peer is given an assigned local IP address to access the inside network. However, you might want to translate the local IP address back to the peer’s real public IP address if, for example, your inside servers and network security is based on the peer’s real IP address.

You can enable this feature on one interface per tunnel group. Object NAT rules are dynamically added and deleted when the VPN session is established or disconnected. You can view the rules using the `show nat` command.

*Note* Because of routing issues, we do not recommend using this feature unless you know you need this feature; contact Cisco TAC to confirm feature compatibility with your network. See the following limitations:

- Only supports Cisco IPsec and AnyConnect Client.
- Return traffic to the public IP addresses must be routed back to the ASA 1000V so the NAT policy and VPN policy can be applied.
- Does not support load-balancing (because of routing issues).
- Does not support roaming (public IP changing).

We introduced the following command: `nat-assigned-to-public-ip interface (tunnel-group general-attributes configuration mode)`.
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 13-1
- Prerequisites for Twice NAT, page 13-2
- Guidelines and Limitations, page 13-2
- Default Settings, page 13-2
- Configuring Twice NAT, page 13-3
- Monitoring Twice NAT, page 13-24
- Configuration Examples for Twice NAT, page 13-24
- Feature History for Twice NAT, page 13-28

Note: For detailed information about how NAT works, see Chapter 11, “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.
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 8-1.
- 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 8-3.

Guidelines and Limitations

- 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, use the `interface` keyword instead of the IP address.
  - (Dynamic NAT) The standby interface IP address when VPN is enabled.
- You cannot configure interface PAT on the inside security profile interfaces, because they do not have IP addresses.

Default Settings

- By default, the rule is added to the end of section 1 of the NAT table.
- 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 ASA 1000V 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 13-3
- Configuring Dynamic PAT (Hide), page 13-8
- Configuring Static NAT or Static NAT-with-Port-Translation, page 13-15
- Configuring Identity NAT, page 13-20

Configuring Dynamic NAT

This section describes how to configure twice NAT for dynamic NAT. For more information, see the “Dynamic NAT” section on page 11-9.
## 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 real source addresses.</td>
</tr>
</tbody>
</table>
| ```
object network obj_name
   {host ip_address | subnet subnet_address netmask | range ip_address_1 ip_address_2}
``` | You can configure either a network object or a network object group. For more information, see the “Configuring Objects” section on page 8-3. |
| Network object group: | If you want to translate all traffic, you can skip this step and specify the any keyword instead of creating an object or 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 MyInsNet | | |
| hostname(config-network-object)# subnet | | |
| 10.1.1.0 255.255.255.0 | | |
| **Step 2** | | |
| Network object: | Configure the mapped source addresses. |
| ```
object network obj_name
   range ip_address_1 ip_address_2
``` | You can configure either a network object or a network object group. |
| Network object group: | 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. |
| ```
object-group network grp_name
   {network-object {object net_obj_name | host ip_address} | group-object grp_obj_name}
``` | **Note**  The mapped object or group cannot contain a subnet. |
| Example: | See the “Guidelines and Limitations” section on page 13-2 for information about disallowed mapped IP addresses. |
| hostname(config)# object network NAT_POOL | | |
| hostname(config-network-object)# range | | |
| 209.165.201.10 209.165.201.20 | | |
### Chapter 13  Configuring Twice NAT

#### Configuring Twice NAT

**Step 3**

(Optional)

Network object:

```plaintext
object network obj_name
    (host ip_address | subnet subnet_address netmask | range ip_address_1 ip_address_2)
```

Network object group:

```plaintext
object-group network grp_name
    (network-object (object net_obj_name | subnet_address netmask | host ip_address) | group-object grp_obj_name)
```

**Example:**

```plaintext
hostname(config)# object network Server1
hostname(config-network-object)# host 209.165.201.8
```

**Step 4**

(Optional)

Network object:

```plaintext
object network obj_name
    (host ip_address | subnet subnet_address netmask | range ip_address_1 ip_address_2)
```

Network object group:

```plaintext
object-group network grp_name
    (network-object (object net_obj_name | subnet_address netmask | host ip_address) | group-object grp_obj_name)
```

**Example:**

```plaintext
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.</td>
</tr>
<tr>
<td>(Optional) Network object:</td>
<td>You can configure either a network object or a network object group.</td>
</tr>
</tbody>
</table>
| ```plaintext
object network obj_name
    (host ip_address | subnet subnet_address netmask | range ip_address_1 ip_address_2)
``` | 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 11-14. |
| **Step 4** | Configure the **mapped destination** addresses. |
| (Optional) Network object: | 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. |
| ```plaintext
object network obj_name
    (host ip_address | subnet subnet_address netmask | range ip_address_1 ip_address_2)
``` | 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 11-4. |
| **Example:** | For static interface NAT with port translation, 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 11-6. |
| ```plaintext
hostname(config)# object network Server1
hostname(config-network-object)# host 10.1.1.67
``` | |
### Step 5

(Optional)

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>object service obj_name</code></td>
<td>Configure service objects for:</td>
</tr>
<tr>
<td>`service (tcp</td>
<td>udp) destination`</td>
</tr>
<tr>
<td><code>operator port</code></td>
<td>• Destination mapped port</td>
</tr>
</tbody>
</table>

**Example:**

```
hostname(config)# object service REAL_SVC
destination eq 80
```

```
hostname(config)# object service MAPPED_SVC
destination eq 8080
```

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.
Step 6

```
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_dest_svc_obj
real_dest_svc_obj] [dns] [inactive]
[|description desc]
```

Configure dynamic NAT. See the following guidelines:

- **Interfaces**—Specify the real and mapped interfaces. Be sure to include the parentheses in your command. 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 11-18). 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—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 the outside interface for the *mapped_ifc*; inside security profile interfaces do not support interface PAT.

Example:

```
hostname(config)# nat {inside,outside}
source dynamic MyInsNet NAT_POOL
destination static Server1_mapped Server1
service MAPPED_SVC REAL_SVC
```
Chapter 13  Configuring Twice NAT

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 11-11.

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. Therefore, ports below 1024 have only a small PAT pool that can be used. If you have a lot of traffic that uses the lower port ranges, you can now specify a flat range of ports to be used instead of the three unequal-sized tiers: either 1024 to 65535, or 1 to 65535.
If you use the same PAT pool object in two separate rules, then be sure to specify the same options for each rule. For example, if one rule specifies extended PAT and a flat range, then the other rule must also specify extended PAT and a flat range.

For extended PAT for a PAT pool:

- Many application inspections do not support extended PAT. See the “Default Settings” section on page 19-3 in Chapter 19, “Getting Started with Application Layer Protocol Inspection,” for a complete list of unsupported inspections.
- If you enable extended PAT for a dynamic PAT rule, then you cannot also use an address in the PAT pool as the PAT address in a separate static NAT-with-port-translation rule. For example, if the PAT pool includes 10.1.1.1, then you cannot create a static NAT-with-port-translation rule using 10.1.1.1 as the PAT address.
- If you use a PAT pool and specify an interface for fallback, you cannot specify extended PAT.
- For VoIP deployments that use ICE or TURN, do not use extended PAT. ICE and TURN rely on the PAT binding to be the same for all destinations.

For round robin for a PAT pool:

- If a host has an existing connection, then subsequent connections from that host will use the same PAT IP address if ports are available. **Note:** This “stickiness” does not survive a failover. If the ASA 1000V fails over, then subsequent connections from a host may not use the initial IP address.
- Round robin, especially when combined with extended PAT, can consume a large amount of memory. Because NAT pools are created for every mapped protocol/IP address/port range, round robin results in a large number of concurrent NAT pools, which use memory. Extended PAT results in an even larger number of concurrent NAT pools.

**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>object network obj_name</td>
<td>You can configure either a network object or a network object group. For more information, see the “Configuring Objects” section on page 8-3.</td>
</tr>
<tr>
<td>host ip_address</td>
<td>If you want to translate all traffic, you can skip this step and specify the <strong>any</strong> keyword instead of creating an object or group.</td>
</tr>
<tr>
<td>subnet subnet_address netmask</td>
<td></td>
</tr>
</tbody>
</table>
## Chapter 13      Configuring Twice NAT

### Step 2

**Network object:**

```plaintext
object network obj_name
  {host ip_address | range ip_address_1 ip_address_2}
```

**Network object group:**

```plaintext
object-group network grp_name
  {network-object {object net_obj_name | host ip_address} | group-object grp_obj_name}
```

**Example:**

```plaintext
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)# object-group network PAT_POOLS
hostname(config-network)# network-object object PAT_POOL1
hostname(config-network)# network-object object PAT_POOL2
hostname(config-network)# network-object object PAT_IP
```

**Step 3** *(Optional)*

**Network object:**

```plaintext
object network obj_name
  {host ip_address | subnet subnet_address netmask | range ip_address_1 ip_address_2}
```

**Network object group:**

```plaintext
object-group network grp_name
  {network-object {object net_obj_name | subnet_address netmask | host ip_address} | group-object grp_obj_name}
```

**Example:**

```plaintext
hostname(config)# object network Server1
hostname(config-network-object)# host 209.165.201.8
```

### Command Purpose

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 2</strong></td>
<td>Specify the <strong>mapped</strong> 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 <code>nat</code> command or if you want to use the interface address by specifying the <code>interface</code> keyword. For mapped addresses used as a PAT pool, all addresses in the object or group, including ranges, are used as PAT addresses. <strong>Note</strong> The object or group cannot contain a subnet. See the “Guidelines and Limitations” section on page 13-2 for information about disallowed mapped IP addresses. For more information about configuring a network object or group, see the “Configuring Objects” section on page 8-3.</td>
</tr>
<tr>
<td><strong>Step 3</strong> <em>(Optional)</em></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 11-14.</td>
</tr>
</tbody>
</table>
## 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
```

## Step 5 (Optional)

**object service obj_name**

```
service (tcp | udp) destination operator port
```

**Example:**

```
hostname(config)# object service REAL_SVC
tcp destination eq 80
```

```
hostname(config)# object service MAPPED_SVC
tcp destination eq 8080
```

---

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 4</strong> (Optional)</td>
<td>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 11-4. For static interface NAT with port translation, 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 11-6.</td>
</tr>
</tbody>
</table>
| **Step 5** (Optional) | 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. |
## Configuring Twice NAT

### Step 6

The `nat` command is used to configure the network address translation (NAT) settings for your Cisco ASA 1000V device. The command syntax is as follows:

```plaintext
nat [(real_ifc,mapped_ifc)]
    [line | {after-auto [line]}]
source dynamic (real-obj | any)
    (mapped_obj [interface] | [pat-pool
    mapped_obj [round-robin] [extended]
    [flat [include-reserve]] [interface] |
    interface) [destination static (mapped_obj
    | interface) real_obj]
    {service mapped_dest_svc_obj
    real_dest_svc_obj} [dns] [inactive]
    [description desc]
```

### Purpose

Configures **dynamic PAT (hide)**. See the following guidelines:

- **Interfaces**—Specify the real and mapped interfaces. Be sure to include the parentheses in your command. 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 11-18). 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**—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 the outside interface for the `mapped_ifc`; inside security profile interfaces do not support interface PAT.

### Example:

```plaintext
hostname(config)# nat (inside, outside)
destination dynamic MyInsNet interface
description static Server1 Server1
destination static Server2 Server2
description Interface PAT for inside addresses when going to server 1
```

(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.

- **Extended PAT**—The `extended` keyword enables extended PAT. Extended PAT uses 65535 ports per `service`, as opposed to per IP address, by including the destination address and port in the translation information. Normally, the destination port and address are not considered when creating PAT translations, so you are limited to 65535 ports per PAT address. For example, with extended PAT, you can create a translation of 10.1.1.1:1027 when going to 192.168.1.7:23 as well as a translation of 10.1.1.1:1027 when going to 192.168.1.7:80.

- **Flat range**—The `flat` keyword enables use of the entire 1024 to 65535 port range when allocating ports. When choosing the mapped port number for a translation, the ASA 1000V uses the real source port number if it is available. However, without this option, 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: 1 to 511, 512 to 1023, and 1024 to 65535. To avoid running out of ports at the low ranges, configure this setting. To use the entire range of 1 to 65535, also specify the `include-reserve` keyword.

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(continued)</td>
</tr>
<tr>
<td></td>
<td>(continued)</td>
</tr>
</tbody>
</table>

(continued)
### Command

<table>
<thead>
<tr>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>(continued)</td>
</tr>
<tr>
<td>• Destination addresses (Optional):</td>
</tr>
<tr>
<td>- Mapped—Specify a network object or group, or for static interface NAT with port translation only, specify the <code>interface</code> keyword (see Step 4). If you specify <code>interface</code>, be sure to also configure the <code>service</code> keyword. For this option, you must configure the outside interface for the <code>mapped_ifc</code>; inside security profile interfaces do not support interface PAT. See the “Static Interface NAT with Port Translation” section on page 11-6 for more information.</td>
</tr>
<tr>
<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>
<tr>
<td>• Destination port—(Optional) Specify the <code>service</code> 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.</td>
</tr>
<tr>
<td>• DNS—(Optional; for a source-only rule) The <code>dns</code> keyword translates DNS replies. Be sure DNS inspection is enabled (it is enabled by default). You cannot configure the <code>dns</code> keyword if you configure a <code>destination</code> address. See the “DNS and NAT” section on page 11-22 for more information.</td>
</tr>
<tr>
<td>• Inactive—(Optional) To make this rule inactive without having to remove the command, use the <code>inactive</code> keyword. To reactivate it, reenter the whole command without the <code>inactive</code> keyword.</td>
</tr>
<tr>
<td>• Description—(Optional) Provide a description up to 200 characters using the <code>description</code> keyword.</td>
</tr>
</tbody>
</table>
## 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 11-4.

### 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 real source addresses.</td>
</tr>
</tbody>
</table>
| object network obj_name  
(host ip_address | subnet  
subnet_address netmask | range  
ip_address_1 ip_address_2) | You can configure either a network object or a network object group. For more information, see the “Configuring Objects” section on page 8-3. |
| 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 MyInsNet  
hostname(config-network-object)# subnet 10.1.1.0 255.255.255.0 | | 
| **Step 2** | | 
| Network object: | Configure the mapped source addresses. |
| object network obj_name  
(host ip_address | subnet  
subnet_address netmask | range  
ip_address_1 ip_address_2) | 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 11-4. |
| 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) | For static interface NAT with port translation, 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 11-6. |
| **Example:** | See the “Guidelines and Limitations” section on page 13-2 for information about disallowed mapped IP addresses. |
| hostname(config)# object network MyInsNet_mapped  
hostname(config-network-object)# subnet 192.168.1.0 255.255.255.0 | |
Chapter 13      Configuring Twice NAT

Step 3  
(Optionalal)

Network object:

\[ \text{object network obj} \_\text{name} \]
\[
\{ \text{host ip} \_\text{address} | \text{subnet subnet} \_\text{address netmask} | \text{range ip} \_\text{address} \_1 \text{ip} \_\text{address} \_2 \}
\]

Network object group:

\[ \text{object-group network grp} \_\text{name} \]
\[
\{ \text{network-object} \{ \text{object net} \_\text{obj} \_\text{name} | \text{host ip} \_\text{address} | \text{group-object grp_obj} \_\text{name} \}
\]

Example:

hostname(config)# object network Server1
hostname(config-network-object)# host 209.165.201.8

Step 4  
(Optionalal)

Network object:

\[ \text{object network obj} \_\text{name} \]
\[
\{ \text{host ip} \_\text{address} | \text{subnet subnet} \_\text{address netmask} | \text{range ip} \_\text{address} \_1 \text{ip} \_\text{address} \_2 \}
\]

Network object group:

\[ \text{object-group network grp} \_\text{name} \]
\[
\{ \text{network-object} \{ \text{object net} \_\text{obj} \_\text{name} | \text{host ip} \_\text{address} | \text{group-object grp_obj} \_\text{name} \}
\]

Example:

hostname(config)# object network Server1\_mapped
hostname(config-network-object)# host 10.1.1.67

Command | Purpose
---|---
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 11-14.

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 11-4.
For static interface NAT with port translation, 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 11-6.
### 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
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configure service objects for:</td>
<td></td>
</tr>
<tr>
<td>• Source or destination real port</td>
<td></td>
</tr>
<tr>
<td>• Source or destination mapped port</td>
<td></td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Step 6</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| `nat [(real_ifc,mapped_ifc)]
   [line | {after-object [line]})
   source static real_ob
   [mapped_obj | interface]
   [destination static (mapped_obj | interface) real_obj]
   [service real_src_mapped_dest_svc_obj
    mapped_src_real_dest_svc_obj] [dns]
   [no-proxy-arp] [inactive]
   [description desc] | Configures static NAT. See the following guidelines: |

#### Interfaces—Specify the real and mapped interfaces. Be sure to include the parentheses in your command. 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 11-18 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:

- **Real**—Specify a network object or group (see Step 1).
- **Mapped**—Specify a different network object or group (see Step 2). For static interface NAT with port translation only, you can specify the `interface` keyword. If you specify `interface`, be sure to also configure the `service` keyword (in this case, the service objects should include only the source port). For this option, you must configure the outside interface for the `mapped_ifc`; inside security profile interfaces do not support interface PAT. See the “Static Interface NAT with Port Translation” section on page 11-6 for more information.

#### Destination addresses (Optional):

- **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 the outside interface for the `mapped_ifc`; inside security profile interfaces do not support interface PAT.
- **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.

Example:

```bash
hostname(config)# nat (inside,dmz) source static MyInsNet MyInsNet_mapped
destination static Server1 Server1 service REAL_SRC_SVC MAPPED_SRC_SVC
```
### Command | Purpose
--- | ---
(Continued) |  
- **Ports**—(Optional) Specify the `service` keyword along with the real and mapped service objects (see Step 5). 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 mapped_obj 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).
- **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 11-22 for more information.
- **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 11-20 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.

### 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 11-12.

**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>
</tbody>
</table>
| object network **obj_name**  
  {host ip_address | subnet  
  subnet_address netmask | range  
  ip_address_1 ip_address_2} | Configure the real source addresses. |
| **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} | You can configure either a network object or a network object group. For more information, see the “Configuring Objects” section on page 8-3. 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
```
Chapter 13      Configuring Twice NAT

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

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 11-14.

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

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 11-4.

For static interface NAT with port translation, 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 11-6.
### Chapter 13      Configuring Twice NAT

#### Step 4
(Optional)

```plaintext
object service obj_name
  service {tcp | udp} [source operator port] [destination operator port]
```

**Example:**

```plaintext
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.

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>object service obj_name</strong></td>
<td>Configure service objects for:</td>
</tr>
<tr>
<td>**service {tcp</td>
<td>udp} [source operator port] [destination operator port]**</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>hostname(config)# object service REAL_SRC_SVC</td>
<td></td>
</tr>
<tr>
<td>hostname(config-service-object)# service tcp source eq 80</td>
<td></td>
</tr>
<tr>
<td>hostname(config)# object service MAPPED_SRC_SVC</td>
<td></td>
</tr>
<tr>
<td>hostname(config-service-object)# service tcp source eq 8080</td>
<td></td>
</tr>
</tbody>
</table>
### Command

#### Step 5

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>nat (real_ifc,mapped_ifc)</code></td>
<td>Configures identity NAT. See the following guidelines:</td>
</tr>
<tr>
<td>`[line</td>
<td>{after-object</td>
</tr>
<tr>
<td>`source static (nw_obj nw_obj</td>
<td>any any)`</td>
</tr>
<tr>
<td>`[destination static</td>
<td>mapped_obj</td>
</tr>
<tr>
<td><code>[service real_src_mapped_dest_svc_obj mapped_src_real_dest_svc_obj]</code></td>
<td>• Destination addresses (Optional):</td>
</tr>
<tr>
<td><code>[no-proxy-arp] [route-lookup] [inactive] [description desc]</code></td>
<td>− Mapped—Specify a network object or group, or for static interface NAT with port translation only, specify the <code>interface</code> keyword (see Step 3). If you specify <code>interface</code>, be sure to also configure the <code>service</code> keyword (in this case, the service objects should include only the destination port). For this option, you must configure the outside interface for the <code>mapped_ifc</code>; inside security profile interfaces do not support interface PAT. See the “Static Interface NAT with Port Translation” section on page 11-6 for more information.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>− 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.</td>
</tr>
</tbody>
</table>

```bash
hostname(config)# nat (inside,outside)
source static MyInsNet MyInsNet
destination static Server1 Server1
```

• 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 mapped_obj 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).
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>

(Continued)

- 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 11-20 for more information.
- Route lookup—(Optional; interface(s) specified) Specify route-lookup to determine the egress interface using a route lookup instead of using the interface specified in the NAT command. See the “Determining the Egress Interface” section on page 11-22 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.

**Monitoring Twice NAT**

To monitor twice NAT, enter one of the following commands:

**Configuration Examples for Twice NAT**

This section includes the following configuration examples:

- Different Translation Depending on the Destination (Dynamic PAT), page 13-24
- Different Translation Depending on the Destination Address and Port (Dynamic PAT), page 13-26

**Different Translation Depending on the Destination (Dynamic PAT)**

Figure 13-1 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.
### Step 1
Add a network object for the inside network:

```bash
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 outside network 1:

```bash
hostname(config)# object network outsideNetwork1
hostname(config-network-object)# subnet 209.165.201.0 255.255.255.224
```

### Step 3
Add a network object for the PAT address:

```bash
hostname(config)# object network PATaddress1
hostname(config-network-object)# host 209.165.202.129
```

### Step 4
Configure the first twice NAT rule:

```bash
hostname(config)# nat (Mktg,outside) source dynamic myInsideNetwork PATaddress1
destination static outsideNetwork1 outsideNetwork1
```

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 13-8 for more information about specifying the section and line number for the NAT rule.

### Step 5
Add a network object for the outside network 2:

```bash
hostname(config)# object network outsideNetwork2
hostname(config-network-object)# subnet 209.165.200.224 255.255.255.224
```

### Step 6
Add a network object for the PAT address:

```bash
hostname(config)# object network PATaddress2
```
**Step 7**  Configure the second twice NAT rule:

```plaintext
hostname(config)# nat (Mktg,outside) source dynamic myInsideNetwork PATaddress2
    destination static outsideNetwork2 outsideNetwork2
```

## Different Translation Depending on the Destination Address and Port (Dynamic PAT)

*Figure 13-2* 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 13-2  Twice NAT with Different Destination Ports*

<table>
<thead>
<tr>
<th>Sec Profile</th>
<th>Translation</th>
<th>Web Packet Dest. Address:</th>
<th>Telnet Packet Dest. Address:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mktg</td>
<td>10.1.2.27:80</td>
<td>209.165.201.11:80</td>
<td>10.1.2.27</td>
</tr>
<tr>
<td></td>
<td>209.165.201.11:80</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:

```plaintext
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:

```plaintext
hostname(config)# object network TelnetWebServer
hostname(config-network-object)# host 209.165.201.11
```

**Step 3**  Add a network object for the PAT address when using Telnet:

```plaintext
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 (Mktg,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 13-8 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 (Mktg,outside) source dynamic myInsideNetwork PATaddress2
destination static TelnetWebServer TelnetWebServer service HTTPObj HTTPObj
```
# Feature History for Twice NAT

Table 13-1 lists each feature change and the platform release in which it was implemented.

## Table 13-1  Feature History for Twice NAT

<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. We modified or introduced the following commands: nat, 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. 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>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: <code>nat source dynamic [pat-pool mapped_object [round-robin]]</code>.</td>
</tr>
</tbody>
</table>
### Table 13-1  Feature History for Twice NAT (continued)

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
</table>
| Round robin PAT pool allocation uses the same IP address for existing hosts | 8.4(3)            | When using a PAT pool with round robin allocation, if a host has an existing connection, then subsequent connections from that host will use the same PAT IP address if ports are available.  
We did not modify any commands.  
This feature is not available in 8.5(1) or 8.6(1). |
### Feature History for Twice NAT

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extended PAT for a PAT pool</td>
<td>8.4(3)</td>
<td>Each PAT IP address allows up to 65535 ports. If 65535 ports do not provide enough translations, you can now enable extended PAT for a PAT pool. Extended PAT uses 65535 ports per service, as opposed to per IP address, by including the destination address and port in the translation information. We modified the following commands: <code>nat source dynamic [pat-pool mapped_object [extended]].</code> This feature is not available in 8.5(1) or 8.6(1).</td>
</tr>
</tbody>
</table>
| Automatic NAT rules to translate a VPN peer’s local IP address back to the peer’s real IP address | 8.4(3)            | In rare situations, you might want to use a VPN peer’s real IP address on the inside network instead of an assigned local IP address. Normally with VPN, the peer is given an assigned local IP address to access the inside network. However, you might want to translate the local IP address back to the peer’s real public IP address if, for example, your inside servers and network security is based on the peer’s real IP address. You can enable this feature on one interface per tunnel group. Object NAT rules are dynamically added and deleted when the VPN session is established or disconnected. You can view the rules using the `show nat` command. Note Because of routing issues, we do not recommend using this feature unless you know you need this feature; contact Cisco TAC to confirm feature compatibility with your network. See the following limitations:  
  - Only supports Cisco IPsec and AnyConnect Client.  
  - Return traffic to the public IP addresses must be routed back to the ASA 1000V so the NAT policy and VPN policy can be applied.  
  - Does not support load-balancing (because of routing issues).  
  - Does not support roaming (public IP changing).  
We introduced the following command: `nat-assigned-to-public-ip interface` (tunnel-group general-attributes configuration mode). |
PART 5

Configuring Service Policies Using the Modular Policy Framework
CHAPTER 14

Configuring a Service Policy Using the Modular Policy Framework

Service policies using Modular Policy Framework provide a consistent and flexible way to configure ASA 1000V 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 14-1
- Guidelines and Limitations, page 14-6
- Default Settings, page 14-6
- Task Flows for Configuring Service Policies, page 14-8
- Identifying Traffic (Layer 3/4 Class Maps), page 14-10
- Defining Actions (Layer 3/4 Policy Map), page 14-14
- Applying Actions to an Interface (Service Policy), page 14-16
- Monitoring Modular Policy Framework, page 14-17
- Configuration Examples for Modular Policy Framework, page 14-17
- Feature History for Service Policies, page 14-20

Information About Service Policies

This section describes how service policies work and includes the following topics:

- Supported Features for Through Traffic, page 14-2
- Supported Features for Management Traffic, page 14-2
- Feature Directionality, page 14-2
- Feature Matching Within a Service Policy, page 14-3
- Order in Which Multiple Feature Actions are Applied, page 14-3
- Incompatibility of Certain Feature Actions, page 14-4
- Feature Matching for Multiple Service Policies, page 14-5
Supported Features for Through Traffic

Table 14-1 lists the features supported by Modular Policy Framework.

**Table 14-1  Modular Policy Framework**

<table>
<thead>
<tr>
<th>Feature</th>
<th>See:</th>
</tr>
</thead>
</table>
| Application inspection (multiple types)           | • Chapter 19, “Getting Started with Application Layer Protocol Inspection.”  
|                                                   | • Chapter 20, “Configuring Inspection of Basic Internet Protocols.”    |
|                                                   | • Chapter 22, “Configuring Inspection of Database and Directory Protocols.” |
|                                                   | • Chapter 23, “Configuring Inspection for Management Application Protocols.” |
| TCP and UDP connection limits and timeouts, and TCP sequence number randomization | Chapter 24, “Configuring Connection Settings.”                          |
| TCP normalization                                 | Chapter 24, “Configuring Connection Settings.”                          |
| TCP state bypass                                  | Chapter 24, “Configuring Connection Settings.”                          |

**Supported Features for Management Traffic**

Modular Policy Framework supports the following features for management traffic:

- Application inspection for RADIUS accounting traffic—See Chapter 23, “Configuring Inspection for Management Application Protocols.”
- Connection limits—See Chapter 24, “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, only traffic that enters (or exits, depending on the feature) the interface to which you apply the policy map is affected. See Table 14-2 for the directionality of each feature.

**Table 14-2 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>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 ASA 1000V 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 ASA 1000V also applies the actions for the subsequent class map, if supported. See the “Incompatibility of Certain Feature Actions” section on page 14-4 for more information about unsupported combinations.

For example, if a packet matches a class map for connection limits, and also matches a class map for 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.

**Note** Application inspection includes multiple inspection types, and each inspection type is a separate feature when you consider the matching guidelines above.

**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.

Actions are performed in the following order:

1. TCP normalization, TCP and UDP connection limits and timeouts, TCP sequence number randomization, and TCP state bypass.

**Note** When a the ASA 1000V performs a proxy service (such as AAA) 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.

2. Application inspection (multiple types)
The order of application inspections applied when a class of traffic is classified for multiple inspections is as follows. Only one inspection type can be applied to the same traffic. WAAS inspection is an exception, because it can be applied along with other inspections for the same traffic. See the “Incompatibility of Certain Feature Actions” section on page 14-4 for more information.

a. CTIQBE  
b. DNS  
c. FTP  
d. H323  
e. HTTP  
f. ICMP  
g. ICMP error  
h. ILS  
i. MGCP  
j. NetBIOS  
k. PPTP  
l. Sun RPC  
m. RSH  
n. RTSP  
o. SIP  
p. Skinny  
q. SMTP  
r. SNMP  
s. SQL*Net  
t. TFTP  
u. XDMCP  
v. DCERPC  
w. Instant Messaging  

**Note**  
RADIUS accounting is not listed because it is the only inspection allowed on management traffic. WAAS is not listed because it can be configured along with other inspections for the same traffic.

**Incompatibility of Certain Feature Actions**

Some features are not compatible with each other for the same traffic. Also, most inspections should not be combined with another inspection, so the ASA 1000V only applies one inspection if you configure multiple inspections for the same traffic. In this case, the feature that is applied is the higher priority feature in the list in the “Order in Which Multiple Feature Actions are Applied” section on page 14-3.

For information about compatibility of each feature, see the chapter or section for your feature.
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 ASA 1000V, then the ASA 1000V applies the TFTP inspection; when TCP traffic for port 21 arrives, then the ASA 1000V applies the FTP inspection. So in this case only, you can configure multiple inspections for the same class map. Normally, the ASA 1000V 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 14-1, traffic destined to port 21 is mistakenly configured for both FTP and HTTP inspection. In Example 14-2, 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 14-1 Misconfiguration for FTP packets: HTTP Inspection Also Configured**

```
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 14-2 Misconfiguration for HTTP packets: FTP Inspection Also Configured**

```
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 a security profile 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 security profile interface.
Guidelines and Limitations

Class Map Guidelines
The maximum number of class maps of all types is 255. 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 14-8.

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 14-4.

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 14-7
- Default Class Maps, page 14-8
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
- RTSP
- ESMTMP
- SQLnet
- Skinny (SCCP)
- SunRPC
- XDMCP
- SIP
- NetBIOS
- TFTP
- IP Options

The default policy configuration includes the following commands:

```
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 esmtmp
    inspect sqlnet
    inspect skinny
    inspect sunrpc
    inspect xdmcp
    inspect sip
    inspect netbios
    inspect tftp
    inspect ip-options
  service-policy global_policy global
```
Default Class Maps

The configuration includes a default Layer 3/4 class map that the ASA 1000V 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 ASA 1000V, then the ASA 1000V applies the TFTP inspection; when TCP traffic for port 21 arrives, then the ASA 1000V applies the FTP inspection. So in this case only, you can configure multiple inspections for the same class map. Normally, the ASA 1000V 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.

```
class-map inspection_default
match default-inspection-traffic
```

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 ASA 1000V 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.

```
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 14-8
- Identifying Traffic (Layer 3/4 Class Maps), page 14-10

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 ASA 1000V; or you might only want to perform certain actions on traffic from 10.1.1.0/24 to any destination address.
Chapter 14 Configuring a Service Policy Using the Modular Policy Framework

Task Flows for Configuring Service Policies

See the “Identifying Traffic (Layer 3/4 Class Maps)” section on page 14-10.

**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 15-2 and the “Identifying Traffic in an Inspection Class Map” section on page 15-4.

**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 8-11 and the “Creating a Regular Expression Class Map” section on page 8-13.

**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.
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:


See the “Defining Actions (Layer 3/4 Policy Map)” section on page 14-14 and the “Applying Actions to an Interface (Service Policy)” section on page 14-16.
## 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></td>
<td></td>
</tr>
<tr>
<td><code>class-map class_map_name</code></td>
<td>Creates a Layer 3/4 class map, where <em>class_map_name</em> 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><strong>Step 2</strong></td>
<td>(Optional)</td>
</tr>
<tr>
<td><code>description string</code></td>
<td>Adds a description to the class map.</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>Match traffic using one of the following:</td>
</tr>
<tr>
<td><code>match any</code></td>
<td>Matches all traffic.</td>
</tr>
<tr>
<td><code>match access-list access_list_name</code></td>
<td>Matches traffic specified by an extended access list.</td>
</tr>
<tr>
<td>`match port (tcp</td>
<td>udp) (eq port_num</td>
</tr>
<tr>
<td><strong>Tip</strong></td>
<td>For applications that use multiple, non-contiguous ports, use the <code>match access-list</code> command and define an ACE to match each port.</td>
</tr>
</tbody>
</table>
### Chapter 14 Configuring a Service Policy Using the Modular Policy Framework

#### Identifying Traffic (Layer 3/4 Class Maps)

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **match default-inspection-traffic** | Matches default traffic for inspection: the default TCP and UDP ports used by all applications that the ASA 1000V 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 ASA 1000V, then the ASA 1000V applies the TFTP inspection; when TCP traffic for port 21 arrives, then the ASA 1000V 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” section on page 14-4 for more information about combining actions). Normally, the ASA 1000V 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.

See the “Default Settings” section on page 19-3 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 ASA 1000V performance can be impacted. |
| match dscp value1 [value2] [...] [value8] | Matches DSCP value in an IP header, up to eight DSCP values. |
| Example: hostname(config-cmap)# match dscp af43 cs1 ef | |
| match precedence value1 [value2] [value3] [value4] | Matches up to four precedence values, represented by the TOS byte in the IP header, where value1 through value4 can be 0 to 7, corresponding to the possible precedences. |
| Example: hostname(config-cmap)# match precedence 1 4 | |
Chapter 14  Configuring a Service Policy Using the Modular Policy Framework

Identifying Traffic (Layer 3/4 Class Maps)

Examples

The following is an example for the class-map command:

```
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 ASA 1000V, 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 14-2.

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>match rtp starting_port range</td>
<td>Matches RTP traffic, where the starting_port specifies an even-numbered UDP destination port between 2000 and 65534. The range specifies the number of additional UDP ports to match above the starting_port, between 0 and 16383.</td>
</tr>
<tr>
<td>match tunnel-group name (Optional) match flow ip destination-address</td>
<td>Matches VPN tunnel group traffic. You can also specify one other match command to refine the traffic match. You can specify any of the preceding commands, except for the match any, match access-list, or match default-inspection-traffic commands. Or you can also enter the match flow ip destination-address command to match flows in the tunnel group going to each IP address.</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></td>
</tr>
<tr>
<td>class-map type management class_map_name</td>
<td>Creates a management 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><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td>(Optional) description string</td>
<td>Adds a description to the class map.</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td></td>
</tr>
<tr>
<td>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 access-list access_list_name</td>
<td>Matches traffic specified by an extended access list.</td>
</tr>
<tr>
<td>match port (tcp</td>
<td>udp) (eq port_num</td>
</tr>
<tr>
<td>range port_num port_num)</td>
<td>Tip For applications that use multiple, non-contiguous ports, use the <code>match access-list</code> command and define an ACE to match each port.</td>
</tr>
</tbody>
</table>

---

**Example:**

```plaintext
hostname(config)# class-map type management all_mgmt
```

```plaintext
hostname(config-cmap)# description All management traffic
```

```plaintext
hostname(config-cmap)# match access-list udp
```

```plaintext
hostname(config-cmap)# match tcp eq 80
```
**Detailed Steps**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><strong>policy-map</strong> <em>policy_map_name</em></td>
</tr>
<tr>
<td></td>
<td>Adds the policy map. The <em>policy_map_name</em> 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.</td>
</tr>
<tr>
<td>Example:</td>
<td><strong>hostname(config)# policy-map global_policy</strong></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><em>(Optional)</em></td>
</tr>
<tr>
<td></td>
<td><strong>class</strong> <em>class_map_name</em></td>
</tr>
<tr>
<td></td>
<td>Specifies a previously configured Layer 3/4 class map, where the <em>class_map_name</em> is the name of the class map. See the “Identifying Traffic (Layer 3/4 Class Maps)” section on page 14-10 to add a class map.</td>
</tr>
<tr>
<td>Example:</td>
<td><strong>hostname(config-pmap)# description global policy map</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong> If there is no <strong>match default-inspection-traffic</strong> command in a class map, then at most one <strong>inspect</strong> command is allowed to be configured under the class.</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>Specify one or more actions for this class map.</td>
</tr>
<tr>
<td></td>
<td>See the “Supported Features for Through Traffic” section on page 14-2.</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>Repeat <strong>Step 2</strong> and <strong>Step 3</strong> 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 global 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)# class-map ftp_traffic`  
`hostname(config-cmap)# match port tcp eq 21`
hostname(config)# class-map tcp_traffic
hostname(config-cmap)# match port tcp range 1 65535
hostname(config)# 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)# 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)# 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 ASA 1000V 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 default 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:

```
service-policy 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>
<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>

#### 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 ASA 1000V 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 to HTTP Traffic, page 14-17
- Applying Inspection to HTTP Traffic Globally, page 14-18
- Applying Inspection and Connection Limits to HTTP Traffic to Specific Servers, page 14-19
- Applying Inspection to HTTP Traffic with NAT, page 14-20

Applying Inspection to HTTP Traffic

In this example (see Figure 14-1), any HTTP connection (TCP traffic on port 80) that enters or exits the ASA 1000V through the outside interface is classified for HTTP inspection.

**Figure 14-1 HTTP Inspection**

![HTTP Inspection Diagram]

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 14-2), any HTTP connection (TCP traffic on port 80) that enters the ASA 1000V 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 14-2  Global HTTP Inspection**

![Diagram showing global HTTP inspection](image)

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 14-3), any HTTP connection destined for Server A (TCP traffic on port 80) that enters the ASA 1000V 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 ASA 1000V through the VM1 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 14-3 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 (VM1,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 (VM1,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 VM1
hostname(config)# service-policy policy_serverA interface outside
```
Applying Inspection to HTTP Traffic with NAT

In this example, the host on the VM1 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. 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 also use the real address.

See the following commands for this example:

```
hostname(config)# object network obj-192.168.1.1
hostname(config-network-object)# host 192.168.1.1
hostname(config-network-object)# nat (VM1,outside) static 209.165.200.225

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 VM1
```

Feature History for Service Policies

Table 14-3 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>Configuring Service Policies Using the Modular Policy Framework</td>
<td>8.7(1)</td>
<td>QoS is not supported.</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 15-1
- Guidelines and Limitations, page 15-2
- Default Inspection Policy Maps, page 15-2
- Defining Actions in an Inspection Policy Map, page 15-2
- Identifying Traffic in an Inspection Class Map, page 15-4
- Where to Go Next, page 15-6

Information About Inspection Policy Maps

See the “Configuring Application Layer Protocol Inspection” section on page 19-5 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.
Guidelines and Limitations

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
```

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 ASA 1000V applies the actions is determined by internal ASA 1000V 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.

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
```

### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>(Optional) Create an inspection class map. See the “Identifying Traffic in an Inspection Class Map” section on page 15-4. Alternatively, you can identify the traffic directly within the policy map.</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><code>policy-map type inspect application policy_map_name</code> Creates the inspection policy map. See the “Configuring Application Layer Protocol Inspection” section on page 19-5 for a list of applications that support inspection policy maps. The <code>policy_map_name</code> 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.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><code>hostname(config)# policy-map type inspect ftp ftp_policy</code></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>Specify the traffic on which you want to perform actions using one of the following methods:</td>
</tr>
</tbody>
</table>
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.

### Command

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>class class_map_name</code></td>
<td>Specifies the inspection class map that you created in the “Identifying Traffic in an Inspection Class Map” section on page 15-4. Not all applications support inspection class maps.</td>
</tr>
<tr>
<td><strong>Example:</strong>&lt;br&gt;hostname(config-pmap)# class _traffic&lt;br&gt;hostname(config-pmap-c)#</td>
<td>Specify traffic directly in the policy map using one of the <code>match</code> commands described for each application in the inspection chapter. If you use a <code>match not</code> command, then any traffic that matches the criterion in the <code>match not</code> command does not have the action applied.</td>
</tr>
<tr>
<td><strong>Example:</strong>&lt;br&gt;hostname(config-pmap)# match req-resp&lt;br&gt;content-type mismatch&lt;br&gt;hostname(config-pmap-c)#</td>
<td>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.</td>
</tr>
<tr>
<td>Step 4&lt;br&gt;{drop [send-protocol-error]</td>
<td>drop-connection [send-protocol-error]</td>
</tr>
<tr>
<td><strong>Example:</strong>&lt;br&gt;hostname(config-pmap-c)# drop-connection&lt;br&gt;log</td>
<td>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.</td>
</tr>
<tr>
<td>Step 5&lt;br&gt;parameters</td>
<td>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.</td>
</tr>
<tr>
<td><strong>Example:</strong>&lt;br&gt;hostname(config-pmap)# parameters&lt;br&gt;hostname(config-pmap-p)#</td>
<td></td>
</tr>
</tbody>
</table>
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> (Optional)</td>
<td>Create a regular expression.</td>
</tr>
<tr>
<td>`class-map type inspect application [match-all</td>
<td>match-any] class_map_name`</td>
</tr>
<tr>
<td><strong>Step 3</strong> (Optional)</td>
<td>Adds a description to the class map.</td>
</tr>
<tr>
<td><code>description string</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>Define the traffic to include in the class by entering one or more <code>match</code> commands available for your application.</td>
</tr>
</tbody>
</table>

Examples

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 VM1

Where to Go Next

To use an inspection policy, see Chapter 14, “Configuring a Service Policy Using the Modular Policy Framework.”
PART 6

Configuring Access Control
CHAPTER 16

Configuring Access Rules

This chapter describes how to control network access through the ASA 1000V using access rules and
includes the following sections:

- Information About Access Rules, page 16-1
- Prerequisites, page 16-4
- Default Settings, page 16-4
- Configuring Access Rules, page 16-4
- Monitoring Access Rules, page 16-5
- Configuration Examples for Permitting or Denying Network Access, page 16-5

Note
To access the ASA 1000V 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 17,
“Configuring Management Access.”

Information About Access Rules

You create an access rule by applying an extended access list to an interface or globally for all
interfaces. 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.

This section includes the following topics:

- Implicit Permits, page 16-2
- Implicit Deny, page 16-2
- Inbound and Outbound Rules, page 16-3
- Access Rules for Returning Traffic, page 16-4
- Management Access Rules, page 16-4
How to Apply Access Rules to Interfaces

Access rules applied to the outside interface must refer to the outside Ethernet interface directly. For security policy purposes, the inside interface is divided up into separate security profiles. Access rules applied to the inside interface must refer to a specific security profile, and not the inside interface directly.

![Diagram of ASA 1000V with security profiles]

You can control access between security profiles and the outside, but you cannot control access between security profiles; because hosts defined by security profiles are all on the inside interface, traffic between security profiles does not pass through the ASA 1000V; they can reach each other directly or through the VSG if desired.

Implicit Permits

IPv4 traffic from a security profile interface to the outside interface is allowed through by default. For other traffic access or limitations, you need to use an extended access rule.

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 16-3.

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 ASA 1000V except for particular addresses, then you need to deny the particular addresses and then permit all others.

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 ASA 1000V supports two types of access lists:
- Inbound—Inbound access lists apply to traffic as it enters an interface. Global access rules are always inbound.
- Outbound—Outbound access lists apply to traffic as it exits an interface.

"Inbound" and "outbound" refer to the application of an access list on an interface, either to traffic entering the ASA 1000V on an interface or traffic exiting the ASA 1000V 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 16-1.) The outbound access list prevents any other hosts from reaching the outside network.

See the following commands for this example:
Access Rules for Returning Traffic

For TCP and UDP connections, you do not need an access rule to allow returning traffic because the ASA 1000V allows all returning traffic for established, bidirectional connections.

For connectionless protocols such as ICMP, however, the ASA 1000V 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) (ASA 1000V to host) or `echo` (8) (host to ASA 1000V).

Management Access Rules

You can configure access rules that control management traffic destined to the ASA 1000V. 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.

Prerequisites

Before you can create an access rule, create the access list. See Chapter 9, “Adding an Extended Access List,” for more information.

Default Settings

See the “ImplicitPermits” section on page 16-2.

Configuring Access Rules

To apply an access rule, perform the following steps.
Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **access-group access_list**
| {*(in | out)*} interface interface_name
| *(control-plane) | global)* | Binds an access list to an interface or applies it globally. Specify the extended access list name. 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. To control traffic from inside to outside, specify a security profile interface, not the inside Ethernet interface.
| • 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:

```
show running-config access-group
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>show running-config access-group</strong></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 10.1.1.1. (This IP address is the real address, not the visible one on the outside interface after NAT.)

```
hostname(config)# access-list ACL_OUT extended permit tcp any host 10.1.1.1 eq www
hostname(config)# access-group ACL_OUT in interface outside
```
The following example allows all hosts on security profile interfaces “hr” and “eng” to access the outside network for web traffic. Because the ASA 1000V does not route between eng and hr interfaces, these access-group commands only apply between the security profile interfaces and the outside interface.

```
hostname(config)# access-list ANY extended permit tcp any any eq www
hostname(config)# access-group ANY in interface eng
hostname(config)# access-group ANY in interface hr
```

The following example uses a service object group to permit specific services on the security profile interface “Finance”:

```
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 eq http
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 FinanceAcl extended permit object-group myaclog any any
hostname(config)# access-group FinanceAcl in interface Finance
```
CHAPTER 17

Configuring Management Access

This chapter describes how to access the ASA 1000V for system management through Telnet, SSH, and HTTPS (using ASDM) and how to create login banners.

This chapter includes the following sections:

- Configuring ASA 1000V Access for ASDM, Telnet, or SSH, page 17-1
- Configuring CLI Parameters, page 17-5
- Configuring ICMP Access, page 17-8
- Configuring Management Access Over an IPsec Site-to-Site Tunnel, page 17-10
- Feature History for Management Access, page 17-10

Note

To access the ASA 1000V 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 1000V Access for ASDM, Telnet, or SSH

This section describes how to allow clients to access the ASA 1000V using ASDM, Telnet, or SSH and includes the following topics:

- Guidelines and Limitations, page 17-2
- Configuring Telnet Access, page 17-2
- Using a Telnet Client, page 17-3
- Configuring SSH Access, page 17-3
- Using an SSH Client, page 17-4
- Configuring HTTPS Access for ASDM, page 17-5
Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

- You cannot use Telnet to the lowest security interface unless you use Telnet inside an IPsec site-to-site tunnel.
- Management access to an interface other than the one from which you entered the ASA 1000V 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 only exception to this rule is through an IPsec site-to-site connection. See the “Configuring Management Access Over an IPsec Site-to-Site Tunnel” section on page 17-10.
- The ASA 1000V allows:
  - A maximum of 5 concurrent Telnet connections.
  - A maximum of 5 concurrent SSH connections.
  - A maximum of 5 concurrent ASDM instances.
- The ASA 1000V 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.
- 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 ASA 1000V using Telnet, 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><code>telnet source_IP_address mask source_interface</code></td>
<td>For each address or subnet, identifies the IP addresses from which the ASA 1000V accepts connections. If there is only one interface, you can configure Telnet to access that interface as long as it has a security level of 100.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# telnet 192.168.1.2 255.255.255.255 inside</td>
<td></td>
</tr>
<tr>
<td>Step 2</td>
<td><code>telnet timeout minutes</code></td>
<td>(Optional) Sets the duration for how long a Telnet session can be idle before the ASA 1000V disconnects the session. 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>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# telnet timeout 30</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 ASA 1000V:

```
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 ASA 1000V 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 ASA 1000V 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 18-21), 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 ASA 1000V using SSH, 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><code>crypto key generate rsa modulus</code> modulus_size</td>
<td>(Optional) 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></td>
<td>Example: <code>hostname(config)# crypto key generate rsa modulus 1024</code></td>
<td></td>
</tr>
<tr>
<td>Step 2</td>
<td><code>write memory</code></td>
<td>Saves the RSA keys to persistent flash memory.</td>
</tr>
<tr>
<td></td>
<td>Example: <code>hostname(config)# write memory</code></td>
<td></td>
</tr>
<tr>
<td>Step 3</td>
<td><code>aaa authentication ssh console LOCAL</code></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 18-21 for more information.</td>
</tr>
<tr>
<td>Step 4</td>
<td><code>username username password password</code></td>
<td>Creates a user in the local database that can be used for SSH access.</td>
</tr>
</tbody>
</table>
# Configuring ASA 1000V Access for ASDM, Telnet, or SSH

## Chapter 17      Configuring Management Access

### Configuring ASA 1000V Access for ASDM, Telnet, or SSH

### 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 ASA 1000V:

```
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 ASA 1000V 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 17-3. When starting an SSH session, a dot (.) displays on the ASA 1000V 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 ASA 1000V is busy and has not hung.

### Configuring HTTPS Access for ASDM

To use ASDM, you need to enable the HTTPS server, and allow HTTPS connections to the ASA 1000V. 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 1</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></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 ASA 1000V accepts HTTPS connections.</td>
</tr>
<tr>
<td></td>
<td>Example: <code>hostname(config)# http 192.168.1.2 255.255.255.255 inside</code></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 2</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><code>http server enable [port]</code></td>
<td>(Optional) Enables the HTTPS server. By default, the <code>port</code> is 443. If you change the port number, be sure to include it in the ASDM access URL. For example, if you change the port number to 444, enter the following: <code>https://10.1.1.1:444</code></td>
</tr>
<tr>
<td></td>
<td>Example: <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 192.168.3.0 255.255.255.0 inside
```

### Configuring CLI Parameters

This section includes the following topics:

- **Configuring a Login Banner**, page 17-6
- **Customizing a CLI Prompt**, page 17-7
- **Changing the Console Timeout**, page 17-8
Configuring a Login Banner

You can configure a message to display when a user connects to the ASA 1000V, before a user logs in, or before a user enters privileged EXEC mode.

Restrictions

After a banner is added, Telnet or SSH sessions to ASA 1000V 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, enter the following command:

```
hostname(config)# banner motd Welcome to $(hostname).
hostname(config)# banner motd Contact me at admin@example.com for any issues.
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| `banner {exec | login | motd} text` | Adds a banner to display at one of three times: when a user first connects (message-of-the-day (motd)), when a user logs in (login), and when a user accesses privileged EXEC mode (exec). When a user connects to the ASA 1000V, the message-of-the-day banner appears first, followed by the login banner and prompts. After the user successfully logs in to the ASA 1000V, the exec banner appears.

For the banner text:

- 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 ASA 1000V by including the strings `$(hostname)` and `$(domain)`.

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 ASA 1000V. You can display the following items in the CLI prompt:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</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>
<tr>
<td>priority</td>
<td>Displays the failover priority as pri (primary) or sec (secondary).</td>
</tr>
<tr>
<td>state</td>
<td>Displays the traffic-passing state of the ASA 1000V. The following values appear for the state:</td>
</tr>
<tr>
<td></td>
<td>• act—Failover is enabled, and the ASA 1000V is actively passing traffic.</td>
</tr>
<tr>
<td></td>
<td>• stby—Failover is enabled, and the ASA 1000V is not passing traffic and is in a standby, failed, or another inactive state.</td>
</tr>
<tr>
<td></td>
<td>• actNoFailover—Failover is not enabled, and the ASA 1000V is actively passing traffic.</td>
</tr>
<tr>
<td></td>
<td>• stbyNoFailover—Failover is not enabled, and the ASA 1000V is not passing traffic. This condition might occur when there is an interface failure above the threshold on the standby ASA 1000V.</td>
</tr>
</tbody>
</table>

Detailed Steps

To customize the CLI prompt, enter the following commands:

Example:

```
hostname(config)# prompt hostname
```
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:

**Detailed Steps**

**Command** | **Purpose**
---|---
```text
console timeout number
```
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.

**Example:**
```
hostname(config)# console timeout 0
```

Configuring ICMP Access

By default, you can send ICMP packets to any ASA 1000V interface. This section tells how to limit ICMP management access to the ASA 1000V. You can protect the ASA 1000V from attacks by limiting the addresses of hosts and networks that are allowed to have ICMP access to the ASA 1000V.

**Note**

For allowing ICMP traffic through the ASA 1000V, see Chapter 16, “Configuring Access Rules.”

This section includes the following topics:
- Information About ICMP Access, page 17-8
- Guidelines and Limitations, page 17-9
- Default Settings, page 17-9
- Configuring ICMP Access, page 17-9

Information About ICMP Access

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 ASA 1000V 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 ASA 1000V 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.
Guidelines and Limitations

- The ASA 1000V does not respond to ICMP echo requests directed to a broadcast address.
- The ASA 1000V 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 ASA 1000V interface.

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>`icmp (permit</td>
<td>deny) (host ip_address</td>
</tr>
</tbody>
</table>

Example:
```
hostname(config)# icmp deny host 10.1.1.15 inside
```

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
```
Configuring Management Access Over an IPsec Site-to-Site Tunnel

If your IPsec site-to-site tunnel terminates on one interface, but you want to manage the ASA 1000V by accessing a different interface, you can identify that interface as a management-access interface. For example, if you enter the ASA 1000V from the outside interface, this feature lets you connect to the inside interface using ASDM, SSH, Telnet, or SNMP; or you can ping the inside interface when entering from the outside interface.

This section includes the following topics:
- Guidelines and Limitations, page 17-10
- Configuring the Management Interface, page 17-10

Guidelines and Limitations

You can define only one management access interface.

Configuring the Management Interface

To configure the management interface, enter the following command.

**Detailed Steps**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>management access</code> <code>management_interface</code></td>
<td>The <code>management_interface</code> argument specifies the name of the management interface that you want to access when entering the ASA 1000V from another interface.</td>
</tr>
</tbody>
</table>

**Example:**

```
hostname(config)# management access inside
```

Feature History for Management Access

Table 17-1 lists the feature history.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Management access</td>
<td>8.7(1)</td>
<td>For management access through VPN, only IPsec site-to-site tunnels are supported.</td>
</tr>
</tbody>
</table>
Configuring AAA for Management Access

This chapter describes how to authenticate and authorize users for management access. The chapter includes the following sections:

- Information About AAA for Management Access, page 18-1
- Configuring AAA Servers and Local Users, page 18-10
- Configuring AAA for Management Access, page 18-20
- Monitoring AAA for Management Access, page 18-33
- Additional References, page 18-35
- Feature History for AAA for Management Access, page 18-35

Note

For Telnet, SSH, or ASDM access, first complete the procedures in Chapter 17, “Configuring Management Access.”

Information About AAA for Management Access

AAA enables the ASA 1000V to determine who the user is (authentication), what the user can do (authorization), and what the user did (accounting).

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 18-2
- Information About Authorization, page 18-2
- Information About Accounting, page 18-2
- Summary of Server Support, page 18-5
- RADIUS Server Support, page 18-5
- TACACS+ Server Support, page 18-6
- RSA/SDI Server Support, page 18-6
- NT Server Support, page 18-7
- Kerberos Server Support, page 18-7
- LDAP Server Support, page 18-7
Information About Authentication

Authentication controls access by requiring valid user credentials, which are usually a username and password. You can configure the ASA 1000V to authenticate the following items:

- All administrative connections to the ASA 1000V, including the following sessions:
  - Telnet
  - SSH
  - Serial console
  - ASDM using HTTPS
- The `enable` command

Information About Authorization

Authorization controls access per user after users are authenticated.

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.

The ASA 1000V caches the first 16 authorization requests per user, so if the user accesses the same services during the current authentication session, the ASA 1000V does not resend the request to the authorization server.

Information About Accounting

Accounting tracks traffic that passes through the ASA 1000V, 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 ASA 1000V for the session, the service used, and the duration of each session.

Information About Management Authentication

This section describes authentication for management access and includes the following topics:

- Comparing CLI Access with and without Authentication, page 18-3
- Comparing ASDM Access with and without Authentication, page 18-3
Comparing CLI Access with and without Authentication

How you log into the ASA 1000V 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 or not 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 Configuring Authentication to Access Privileged EXEC Mode (the enable Command), page 18-21), the ASA 1000V 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 18-22 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.

Information About Command Authorization

This section describes command authorization and includes the following topics:

- Supported Command Authorization Methods, page 18-3
- About Preserving User Credentials, page 18-4

Supported Command Authorization Methods

You can use one of two command authorization methods:

- Local privilege levels—Configure the command privilege levels on the ASA 1000V. When a local, RADIUS, or LDAP (if you map LDAP attributes to RADIUS attributes) user authenticates for CLI access, the ASA 1000V 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).
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 ASA 1000V 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 ASA 1000V 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 18-25). (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 ASA 1000V, that user is required to provide a username and password for authentication. The ASA 1000V 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 ASA 1000V.

<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></td>
<td></td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Password</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Summary of Server Support

Table 18-1 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 18-1  Summary of AAA Support

<table>
<thead>
<tr>
<th>AAA Service</th>
<th>Database Type</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Local</td>
</tr>
<tr>
<td>Authentication</td>
<td>Yes</td>
</tr>
<tr>
<td>Authorization</td>
<td>Yes¹</td>
</tr>
<tr>
<td>Accounting</td>
<td>No</td>
</tr>
</tbody>
</table>

1. Local command authorization is supported by privilege level only.
2. Command accounting is available for TACACS+ only.

In addition to the native protocol authentication listed in Table 18-1, the ASA 1000V supports proxying authentication. For example, the ASA 1000V can proxy to an RSA/SDI and/or LDAP server via a RADIUS server.

RADIUS Server Support

The ASA 1000V 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)
- RSA RADIUS in RSA Authentication Manager 5.2, 6.1, and 7.x
- Microsoft

Authentication Methods

The ASA 1000V 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 ASA 1000V 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.
Information About AAA for Management Access

- 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.

**TACACS+ Server Support**

The ASA 1000V 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 18-6
- Two-step Authentication Process, page 18-6
- RSA/SDI Primary and Replica Servers, page 18-7

**RSA/SDI Version Support**

The ASA 1000V 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 ASA 1000V can be either the primary or any one of the replicas. See the “RSA/SDI Primary and Replica Servers” section on page 18-7 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 ASA 1000Vs using the same authentication servers simultaneously. After a successful username lock, the ASA 1000V sends the passcode.

**RSA/SDI Primary and Replica Servers**

The ASA 1000V obtains the server list when the first user authenticates to the configured server, which can be either a primary or a replica. The ASA 1000V 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 ASA 1000V 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 ASA 1000V supports 3DES, DES, and RC4 encryption types.

For a simple Kerberos server configuration example, see Example 18-2 on page 18-15.

**LDAP Server Support**

The ASA 1000V supports LDAP. This section includes the following topics:

- Authentication with LDAP, page 18-7
- LDAP Server Types, page 18-8

**Authentication with LDAP**

During authentication, the ASA 1000V 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 ASA 1000V passes authentication parameters, usually a username and password, to the LDAP server in plain text.

The ASA 1000V supports the following SASL mechanisms, listed in order of increasing strength:

- Digest-MD5—The ASA 1000V responds to the LDAP server with an MD5 value computed from the username and password.
- Kerberos—The ASA 1000V responds to the LDAP server by sending the username and realm using the GSSAPI Kerberos mechanism.

You can configure the ASA 1000V and LDAP server to support any combination of these SASL mechanisms. If you configure multiple mechanisms, the ASA 1000V 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 ASA 1000V and the server. For example, if both the LDAP server and the ASA 1000V support both mechanisms, the ASA 1000V selects Kerberos, the stronger of the mechanisms.

When user LDAP authentication has succeeded, the LDAP server returns the attributes for the authenticated user.

**LDAP Server Types**

The ASA 1000V 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 ASA 1000V 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 ASA 1000V 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 ASA 1000V does not support password management with Novell, OpenLDAP, and other LDAPv3 directory servers.
- The ASA 1000V uses the Login Distinguished Name (DN) and Login Password to establish a trust relationship (bind) with an LDAP server.

**Local Database Support, Including as a Fallback Method**

The ASA 1000V 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 ASA 1000V.

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 ASA 1000V 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 ASA 1000V. 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 ASA 1000V attempts to authenticate to server 1.

If server 1 responds with an authentication failure (such as user not found), the ASA 1000V 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 ASA 1000V tries server 2.

If both servers in the group do not respond, and the ASA 1000V is configured to fall back to the local database, the ASA 1000V tries to authenticate to the local database.

Prerequisites

Prerequisites for Management Authentication

Before the ASA 1000V can authenticate a Telnet, SSH, or HTTP user, you must identify the IP addresses that are allowed to communicate with the ASA 1000V. For more information, see the “Configuring ASA 1000V Access for ASDM, Telnet, or SSH” section on page 17-1.

Prerequisites for Local Command Authorization

- Configure enable authentication. (See the “Configuring Authentication for CLI and ASDM Access” section on page 18-21.)

  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 CVNP3000-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 CVNP3000-Privilege-Level according to the “Configuring LDAP Attribute Maps” section on page 18-15.

Prerequisites for TACACS+ Command Authorization

- Configure CLI authentication (see the “Configuring Authentication for CLI and ASDM Access” section on page 18-21).

- Configure enable authentication (see the “Configuring Authentication to Access Privileged EXEC Mode (the enable Command)” section on page 18-21).
Prerequisites for Management Accounting

- Configure CLI authentication (see the “Configuring Authentication for CLI and ASDM Access” section on page 18-21).
- Configure enable authentication (see the “Configuring Authentication to Access Privileged EXEC Mode (the enable Command)” section on page 18-21).

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 18-28.

Configuring AAA Servers and Local Users

This section includes the following topics:

- Configuring AAA Server Groups, page 18-11
- Configuring LDAP Attribute Maps, page 18-15
- Adding a User Account to the Local Database, page 18-17
- Authenticating Users with a Public Key for SSH, page 18-20

Task Flow for Configuring AAA Servers

Step 1  Do one or both of the following:

- Add a AAA server group. See the “Configuring AAA Server Groups” section on page 18-11.
• Add a user to the local database. See the “Adding a User Account to the Local Database” section on page 18-17.

**Step 2** For an LDAP server, configure LDAP attribute maps. See the “Configuring LDAP Attribute Maps” section on page 18-15.

**Step 3** (Optional) Users can authenticate with a public key. See the “Authenticating Users with a Public Key for SSH” section on page 18-20.

---

### 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.
- Each group can have up to 16 servers.
- 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 ASA 1000V 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 ASA 1000V continues to try the AAA servers.
## Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1**

aaa-server server_tag protocol {kerberos | ldap | nt | radius | sdi | tacacs+}

- **Example:**
  
  hostname(config)# aaa-server servergroup1 protocol ldap
  
  hostname(config-aaa-server-group)#
  
  hostname(config)# aaa-server servergroup1 protocol radius
  
  hostname(config-aaa-server-group)#
  
  hostname(config)# aaa-server servergroup1 protocol radius
  
  hostname(config-aaa-server-group)# ad-agent-mode

  Identifies the server group name and the protocol. For example, to use RADIUS to authenticate network access and TACACS+ to authenticate CLI access, you need to create at least two server groups, one for RADIUS servers and one for TACACS+ servers.

  You can have up to 100 server groups. Each group can have up to 15 servers.

  When you enter the `aaa-server protocol` command, you enter aaa-server group configuration mode.

  The **ad-agent-mode** option specifies the shared secret between the ASA 1000V 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.

  **Step 2**

  max-failed-attempts number

  - **Example:**
    
    hostname(config-aaa-server-group)#
    
    max-failed-attempts 2

  Specifies the maximum number of requests sent to a AAA server in the group before trying the next server. The **number** argument can range from 1 and 5. The default is 3.

  If you configured a fallback method using the local database (for management access only; see the “Configuring Local Command Authorization” section on page 18-25 and the “Configuring TACACS+ Command Authorization” section on page 18-31 to configure the fallback mechanism), and all the servers in the group fail to respond, then the group is considered to be unresponsive, and the fallback method is tried. The server group remains marked as unresponsive for a period of 10 minutes (by default), so that additional AAA requests within that period do not attempt to contact the server group, and the fallback method is used immediately. To change the unresponsive period from the default, see the **reactivation-mode** command in the next step.

  If you do not have a fallback method, the ASA 1000V continues to retry the servers in the group.
### Step 3

**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 `depletion` 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 4

**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 5

**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.

The `interface_name` argument specifies the Ethernet interface name.

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 18-2** 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 18-2  Host Mode Commands, Server Types, and Defaults

<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>kerberos-realm</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>
</tbody>
</table>
Chapter 18      Configuring AAA for Management Access

Configuring AAA Servers and Local Users

Examples

Example 18-1 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 18-1   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

Table 18-2    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-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>
<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 ASA 1000V uses sAMAccountName for LDAP requests. Whether</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>using SASL or plain text, you can secure communications between the ASA</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1000V and the LDAP server with SSL. If you do not configure SASL, we</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>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</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>the server is either a Microsoft, Sun or generic LDAP server, you can</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>manually configure the server type.</td>
</tr>
<tr>
<td>timeout</td>
<td>All</td>
<td>10 seconds</td>
<td></td>
</tr>
</tbody>
</table>
Example 18-2 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 18-2 does not define a retry interval or the port that the Kerberos server listens to, the ASA 1000V uses the default values for these two server-specific parameters. Table 18-2 lists the default values for all AAA server host mode commands.

**Note**
Kerberos realm names use numbers and upper-case letters only. Although the ASA 1000V 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 18-2 Kerberos Server Group and Server**

```plaintext
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
```

**Configuring LDAP Attribute Maps**

The ASA 1000V can use an LDAP directory for authenticating management users.

**Guidelines**

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.
Configuring AAA Servers and Local Users

Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** | **ldap attribute-map map-name**  
**Example:**  
hostname(config)# ldap attribute-map att_map_1 |
|             | Creates an unpopulated LDAP attribute map table. |
| **Step 2** | **map-name user-attribute-name**  
**Cisco-attribute-name**  
**Example:**  
hostname(config-ldap-attribute-map)# map-name department IETF-Radius-Class |
|             | Maps the user-defined attribute name department to the Cisco attribute. |
| **Step 3** | **map-value user-attribute-name**  
**Cisco-attribute-name**  
**Example:**  
hostname(config-ldap-attribute-map)# map-value department Engineering group1 |
|             | Maps the user-defined map value department to the user-defined attribute value and the Cisco attribute value. |
| **Step 4** | **aaa-server server_group [interface_name]**  
**host server_ip**  
**Example:**  
hostname(config)# aaa-server [ldap_dir_1] host 10.1.1.4 |
|             | Identifies the server and the AAA server group to which it belongs.  
The *interface_name* argument specifies the Ethernet interface name. |
| **Step 5** | **ldap-attribute-map map-name**  
**Example:**  
hostname(config-aaa-server-host)# ldap-attribute-map att_map_1 |
|             | Binds the attribute map to the LDAP server. |

Examples

The following example shows how to limit management sessions to the ASA 1000V based on an LDAP attribute called accessType. The accessType attribute has two possible values:

- admin
- helpdesk

The following example shows how each value is mapped to one of the valid IETF-Radius-Service-Type attributes that the ASA 1000V supports: 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 admin 6
hostname(config-ldap-attribute-map)# map-value accessType helpdesk 7
```

```
hostname(config-ldap-attribute-map)# aaa-server LDAP protocol ldap
hostname(config-aaa-server-group)# aaa-server LDAP (inside) host 10.1.254.91
hostname(config-aaa-server-host)# ldap-base-dn CN=Users,DC=cisco,DC=local
```
hostname(config-aaa-server-host)# ldap-scope subtree
hostname(config-aaa-server-host)# ldap-login-password test
hostname(config-aaa-server-host)# ldap-login-dn CN=Administrator,CN=Users,DC=cisco,DC=local
hostname(config-aaa-server-host)# server-type auto-detect
hostname(config-aaa-server-host)# ldap-attribute-map MGMT

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?

ldap mode commands/options:
cisco-attribute-names:
  Access-Hours
  Allow-Network-Extension-Mode
  Auth-Service-Type
  Authenticated-User-Idle-Timeout
  Authorization-Required
  Authorization-Type

  X509-Cert-Data

hostname(config-ldap-attribute-map)#

### Adding a User Account to the Local Database

This section describes how to manage users in the local database.

**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 ASA 1000V 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.
### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> `username username {nopassword</td>
<td>password password} [privilege priv_level]`</td>
</tr>
</tbody>
</table>

**Example:**
```
hostname(config)# username exampleuser1 privilege 1
```

**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. If you want to limit access to privileged EXEC mode, either set the privilege level to 0 or 1, or use the `service-type` command (see Step 4).

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 ASA 1000V 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 DLaUiAX3l78qgoB5c7iVNw== 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 ASA 1000V, and you are using the same password.
## Configuring AAA for Management Access

### Chapter 18

#### Configuring AAA Servers and Local Users

### Step 2

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| ```
hostname(config)# aaa authorization exec authentication-server
``` | (Optional) Enforces user-specific access levels for users who authenticate for management access (see the `aaa authentication console LOCAL` command). This command 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 18-23 for information about configuring a user on a AAA server to accommodate management authorization.

See the following prerequisites for each user type:

- Configure local database users at a privilege level from 0 to 15 using the `username` command. Configure the level of access using the `service-type` command.
- Configure RADIUS users with Cisco VSA CVVPN3000-Privilege-Level with a value between 0 and 15.
- Configure LDAP users with a privilege level between 0 and 15, and then map the LDAP attribute to Cisco VAS CVVPN3000-Privilege-Level using the `ldap map-attributes` command.
- See the `privilege` command for information about setting command privilege levels. |

### Step 3

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| ```
hostname(config-username)# service-type admin | (Optional) Configures the user level if you configured management authorization in Step 2. The `admin` keyword allows full access to any services specified by the `aaa authentication console LOCAL` commands. The `admin` keyword is the default.

The `nas-prompt` keyword allows access to the CLI when you configure the `aaa authentication {telnet | ssh | serial} console LOCAL` command, but denies ASDM configuration access if you configure the `aaa authentication http console LOCAL` command. ASDM monitoring access is allowed. If you enable authentication with the `aaa authentication enable console LOCAL` command, the user cannot access privileged EXEC mode using the `enable` command (or the `login` command).

For more information, see the “Limiting User CLI and ASDM Access with Management Authorization” section on page 18-23.``` | |

---

**Command Purpose**

- `username` command: Configure local database users at a privilege level from 0 to 15.
- `service-type` command: Configure the level of access for local, RADIUS, LDAP, and TACACS+ users.
- `aaa authorization exec LOCAL` command: Enable attributes to be taken from the local database.
- `aaa authentication console LOCAL` command: Configure RADIUS users with CVVPN3000-Privilege-Level.
- `ldap map-attributes` command: Map LDAP attributes to Cisco VAS CVVPN3000-Privilege-Level.
- `privilege` command: Configure command privilege levels.
- `admin` keyword: Full access to any services specified by `aaa authentication console LOCAL` commands.
- `nas-prompt` keyword: Access to the CLI but denies ASDM configuration access if `aaa authentication http console LOCAL` is configured.
- `http console LOCAL` command: Denies ASDM configuration access.
- `enable` command: Disallows access to privileged EXEC mode.
- `login` command: Equivalent to `enable` command.
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:

```
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
```

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]
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| username {user} attributes 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 ASA 1000V is rebooted. |

Configuring AAA for Management Access

- Configuring Authentication for CLI and ASDM Access, page 18-21
- Configuring Authentication to Access Privileged EXEC Mode (the enable Command), page 18-21
- Limiting User CLI and ASDM Access with Management Authorization, page 18-23
- Configuring Command Authorization, page 18-24
## Configuring Authentication for CLI and ASDM Access

To configure management authentication, enter the following command:

### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`aaa authentication {telnet</td>
<td>ssh</td>
</tr>
<tr>
<td>`aaa authentication {telnet</td>
<td>ssh</td>
</tr>
<tr>
<td>`aaa authentication {telnet</td>
<td>ssh</td>
</tr>
<tr>
<td>`aaa authentication {telnet</td>
<td>ssh</td>
</tr>
<tr>
<td>`aaa authentication {telnet</td>
<td>ssh</td>
</tr>
<tr>
<td>`aaa authentication {telnet</td>
<td>ssh</td>
</tr>
</tbody>
</table>

### Configuring Authentication to Access Privileged EXEC Mode (the enable Command)

You can configure the ASA 1000V 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 18-22
- Authenticating Users with the login Command, page 18-22
Configuring Authentication for the enable Command

You can configure the ASA 1000V to authenticate users when they enter the `enable` command. See the “Comparing CLI Access with and without Authentication” section on page 18-3 for more information.

To authenticate users who enter the `enable` command, enter the following command:

```
hostname(config)# aaa authentication enable console LOCAL
```

**Purpose**

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 ASA 1000V 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 ASA 1000V 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 18-25 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:

```
hostname# login
```

**Purpose**

Logs in as a user from the local database. The ASA 1000V prompts for your username and password. After you enter your password, the ASA 1000V places you in the privilege level that the local database specifies.
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.

**Detailed Steps**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>aaa authorization exec authentication-server</td>
</tr>
</tbody>
</table>

**Example:**

```
hostname(config)# aaa authorization exec authentication-server
```

Enables management authorization for local, RADIUS, LDAP (mapped), and TACACS+ users. Also enables support of administrative user privilege levels from RADIUS, which can be used in conjunction with local command privilege levels for command authorization. See the “Configuring Local Command Authorization” section on page 18-25 for more information. Use the `aaa authorization exec LOCAL` command to enable attributes to be taken from the local database.
## Configuring AAA for Management Access

### Configuring Command Authorization

If you want to control access to commands, the ASA 1000V lets you configure command authorization, where you can determine which commands 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

---

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 2</strong></td>
<td>To configure the user for management authorization, see the following requirements for each AAA server type or local user:</td>
</tr>
</tbody>
</table>

- 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). IPsec users can still authenticate and terminate their site-to-site sessions.

Configure Cisco VSA VPN3000-Privilege-Level with a value between 0 and 15. and then map the LDAP attributes to Cisco VSA VPN3000-Privilege-Level using the `ldap map-attributes` command. For more information, see the “Configuring LDAP Attribute Maps” section on page 18-15.

- 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 18-17.
For more information about command authorization, see the “Information About Command Authorization” section on page 18-3.

This section includes the following topics:
- Configuring Local Command Authorization, page 18-25
- Viewing Local Command Privilege Levels, page 18-28
- Configuring Commands on the TACACS+ Server, page 18-29
- Configuring TACACS+ Command Authorization, page 18-31

**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 ASA 1000V 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 18-15.)

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>
</tbody>
</table>

The options in this command are the following:

- **show | clear | cmd**—These optional keywords let you set the privilege only for the show, clear, or configure form of the command. The configure form of the command is typically the form that causes a configuration change, either as the unmodified command (without the show or clear prefix) or as the no form. If you do not use one of these keywords, all forms of the command are affected.

- **level level**—A level between 0 and 15.

- **mode \{enable | configure\**—If a command can be entered in user EXEC or privileged EXEC mode as well as configuration mode, and the command performs different actions in each mode, you can set the privilege level for these modes separately:
  - **enable**—Specifies both user EXEC mode and privileged EXEC mode.
  - **configure**—Specifies configuration mode, accessed using the configure terminal command.

- **command command**—The command you are configuring. You can only configure the privilege level of the main command. For example, you can configure the level of all `aaa` commands, but not the level of the `aaa authentication` command and the `aaa authorization` command separately.

Example:
hostname(config)# privilege show level 5 command filter

Example:
hostname(config)# privilege show level 5 command filter
### Step 2

**aaa authorization exec authentication-server**

**Example:**

```
hostname(config)# aaa authorization exec authentication-server
```

**Purpose:** 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 ASA 1000V 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 18-23 for information about configuring a user on a AAA server to accommodate management authorization.

### Step 3

**aaa authorization command LOCAL**

**Example:**

```
hostname(config)# aaa authorization command LOCAL
```

**Purpose:** Enables the use of local command privilege levels, which can be checked with the privilege level of users in the local database, RADIUS server, or LDAP server (with mapped attributes).

When you set command privilege levels, command authorization does not occur unless you configure command authorization with this command.

### 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
```

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
```
Chapter 18  Configuring AAA for Management Access

Configuring AAA for Management Access

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><code>show running-config all privilege all</code></td>
<td>Shows all commands.</td>
</tr>
<tr>
<td><code>show running-config privilege level level</code></td>
<td>Shows commands for a specific level. The <code>level</code> is an integer between 0 and 15.</td>
</tr>
<tr>
<td><code>show running-config privilege command command</code></td>
<td>Shows the level of a specific command.</td>
</tr>
</tbody>
</table>

**Examples**

For the `show running-config all privilege all` command, the ASA 1000V 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
....
```

The following example displays the command assignments for privilege level 10:

```
hostname(config)# show running-config privilege level 10
privilege show level 10 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 ASA 1000V 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 ASA 1000V 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 a question mark, which shows CLI usage (see **Figure 18-1**).

**Figure 18-1** 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 18-2**).
To disallow some arguments, enter the arguments preceded by \texttt{deny}.

For example, to allow \texttt{enable}, but not \texttt{enable password}, enter \texttt{enable} in the commands field, and \texttt{deny password} in the arguments field. Be sure to check the \texttt{Permit Unmatched Args} check box so that \texttt{enable} alone is still allowed (see Figure 18-3).

When you abbreviate a command at the command line, the ASA 1000V 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 \texttt{sh log}, then the ASA 1000V sends the entire command to the TACACS+ server, \texttt{show logging}. However, if you enter \texttt{sh log mess}, then the ASA 1000V sends \texttt{show logging mess} to the TACACS+ server, and not the expanded command \texttt{show logging message}. You can configure multiple spellings of the same argument to anticipate abbreviations (see Figure 18-4).
We recommend that you allow the following basic commands for all users:

- show checksum
- show curpriv
- enable
- help
- show history
- 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 ASA 1000V 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 ASA 1000V as a user that is defined on the TACACS+ server, and that you have the necessary command authorization to continue configuring the ASA 1000V. 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 ASA 1000V. If you still get locked out, see the “Recovering from a Lockout” section on page 18-33.

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 ASA 1000V. 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 18-24.

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><code>aaa authorization command</code> &lt;br&gt;<code>tacacs+_server_group [LOCAL]</code></td>
<td>Performs command authorization using a TACACS+ server. You can configure the ASA 1000V 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). We recommend that you use the same username and password in the local database as the TACACS+ server because the ASA 1000V prompt does not give any indication of 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 18-17) and command privilege levels (see the “Configuring Local Command Authorization” section on page 18-25).</td>
</tr>
</tbody>
</table>

**Configuring Management Access Accounting**

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>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong>&lt;br&gt;`aaa accounting {serial</td>
<td>telnet</td>
</tr>
</tbody>
</table>

**Example:**

```
hostname(config)# aaa accounting telnet console group_1
```

| **Step 2**<br>`aaa accounting command [privilege level] server-tag` | 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 ASA 1000V should send command accounting messages. |

**Example:**

```
hostname(config)# aaa accounting command privilege 15 group_1
```
Recovering from a Lockout

In some circumstances, when you turn on command authorization or CLI authentication, you can be locked out of the ASA 1000V CLI. You can usually recover access by restarting the ASA 1000V. However, if you already saved your configuration, you might be locked out. Table 18-3 lists the common lockout conditions and how you might recover from them.

Table 18-3 CLI Authentication and Command Authorization Lockout Scenarios

<table>
<thead>
<tr>
<th>Feature</th>
<th>Lockout Condition</th>
<th>Description</th>
<th>Workaround</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>
</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>
</tr>
<tr>
<td>TACACS+ CLI authentication</td>
<td>You are logged in as a user without enough privileges or as a user that does not exist</td>
<td>You enable command authorization, but then find that the user cannot enter any more commands.</td>
<td>Fix the TACACS+ server user account. If you do not have access to the TACACS+ server and you need to configure the ASA 1000V immediately, then log into the maintenance partition and reset the passwords and aaa commands.</td>
</tr>
<tr>
<td>Local command authorization</td>
<td>You are logged in as a user without enough privileges</td>
<td>You enable command authorization, but then find that the user cannot enter any more commands.</td>
<td>Log in and reset the passwords and aaa commands.</td>
</tr>
</tbody>
</table>

Monitoring AAA for Management Access

- Monitoring AAA Servers, page 18-33
- Viewing the Currently Logged-In User, page 18-34

Monitoring AAA Servers

To monitor AAA servers, enter one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show aaa-server</td>
<td>Shows the configured AAA server statistics.</td>
</tr>
<tr>
<td></td>
<td>To clear the AAA server configuration, enter the clear aaa-server</td>
</tr>
<tr>
<td></td>
<td>statistics command.</td>
</tr>
</tbody>
</table>
### Monitoring AAA for Management Access

#### 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 18-4 describes the `show curpriv` command output.

**Table 18-4  show curpriv Command Output Description**

<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 mode) or enable_15 (privileged EXEC mode).</td>
</tr>
<tr>
<td>Current privilege level</td>
<td>Levels range from 0 to 15. Unless you configure local command authorization and assign commands to intermediate privilege levels, levels 0 and 15 are the only levels that are used.</td>
</tr>
<tr>
<td>Current Modes</td>
<td>The available access modes are the following:</td>
</tr>
<tr>
<td></td>
<td>• P_UNPR—User EXEC mode (levels 0 and 1)</td>
</tr>
<tr>
<td></td>
<td>• P_PRIV—Privileged EXEC mode (levels 2 to 15)</td>
</tr>
<tr>
<td></td>
<td>• P_CONF—Configuration mode</td>
</tr>
</tbody>
</table>
Additional References

For additional information related to implementing LDAP mapping, see the next section.

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 for Management Access

Table 18-5 lists the feature history

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>AAA for management access</td>
<td>8.7(1)</td>
<td>For RADIUS or LDAP (mapped) users using the IETF RADIUS Service-Type 5 (Outbound) attribute, remote access (SSL) users are not supported.</td>
</tr>
</tbody>
</table>
PART 7

Configuring Application Inspection
CHAPTER 19

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 ASA 1000V to do 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 ASA 1000V 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 19-1
- Guidelines and Limitations, page 19-3
- Default Settings, page 19-3
- Configuring Application Layer Protocol Inspection, page 19-5

Information about Application Layer Protocol Inspection

This section includes the following topics:

- How Inspection Engines Work, page 19-1
- When to Use Application Protocol Inspection, page 19-2

How Inspection Engines Work

As illustrated in Figure 19-1, the ASA 1000V 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.
Figure 19-1  How Inspection Engines Work

In Figure 19-1, operations are numbered in the order they occur, and are described as follows:

1. A TCP SYN packet arrives at the ASA 1000V to establish a new connection.
2. The ASA 1000V checks the access list database to determine if the connection is permitted.
3. The ASA 1000V creates a new entry in the connection database (XLATE and CONN tables).
4. The ASA 1000V 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 ASA 1000V forwards the packet to the destination system.
6. The destination system responds to the initial request.
7. The ASA 1000V 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 ASA 1000V 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 ASA 1000V 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 ASA 1000V.

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 ASA 1000V 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 ASA 1000V 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.

Failover Guidelines

State information for multimedia sessions that require inspection are not passed over the state link for stateful failover.

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 ASA 1000V 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 ASA 1000V 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 ASA 1000V, there is a best-effort attempt to re-establish a TCP state.

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 19-1 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>
</tbody>
</table>
### Default Settings

**H.323 H.225 and RAS**
- TCP/1720
- UDP/1718
- UDP (RAS) 1718-1719
- No NAT on same security interfaces.
- No static PAT.
- No extended PAT.
- ITU-T H.323, H.245, H225.0, Q.931, Q.932

**HTTP**
- TCP/80
- RFC 2616
- 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.

**ICMP**
- —
- All ICMP traffic is matched in the default class map.

**ICMP ERROR**
- —
- All ICMP traffic is matched in the default class map.

**ILS (LDAP)**
- TCP/389
- No extended PAT.

**Instant Messaging (IM)**
- Varies by client
- No extended PAT.
- RFC 3860

**IP Options**
- —
- RFC 791, RFC 2113
- All IP Options traffic is matched in the default class map.

**MGCP**
- UDP/2427, 2727
- No extended PAT.
- RFC 2705bis-05

**MMP**
- TCP 5443
- No extended PAT.

**NetBIOS Name Server over IP**
- UDP/137, 138 (Source ports)
- No extended PAT.
- NetBIOS is supported by performing NAT of the packets for NBNS UDP port 137 and NBDS UDP port 138.

**PPTP**
- TCP/1723
- RFC 2637

**RADIUS Accounting**
- 1646
- RFC 2865

**RSH**
- TCP/514
- No PAT
- Berkeley UNIX

**RTSP**
- TCP/554
- No extended PAT.
- No outside NAT.
- RFC 2326, 2327, 1889
- No handling for HTTP cloaking.

**SIP**
- TCP/5060
- UDP/5060
- No outside NAT.
- No NAT on same security interfaces.
- No extended PAT.
- RFC 2543

**SKINNY (SCCP)**
- TCP/2000
- No outside NAT.
- No NAT on same security interfaces.
- No extended PAT.
- Does not handle TFTP uploaded Cisco IP Phone configurations under certain circumstances.

**SMTP and ESMTP**
- TCP/25
- RFC 821, 1123

### Table 19-1 Supported Application Inspection Engines (continued)
Chapter 19      Getting Started with Application Layer Protocol Inspection

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 ASA 1000V 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 14, “Configuring a Service Policy Using the Modular Policy Framework.”
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 14-11 and “Creating a Layer 3/4 Class Map for Management Traffic” section on page 14-13 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.

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.

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 ASA 1000V 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 19-3 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
```

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
```
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 23-2
- DNS—See the “Configuring a DNS Inspection Policy Map for Additional Inspection Control” section on page 20-7
- ESMTP—See the “Configuring an ESMTP Inspection Policy Map for Additional Inspection Control” section on page 20-26
- FTP—See the “Configuring an FTP Inspection Policy Map for Additional Inspection Control” section on page 20-12.
- H323—See the “Configuring an H.323 Inspection Policy Map for Additional Inspection Control” section on page 21-6
- Instant Messaging—See the “Configuring an Instant Messaging Inspection Policy Map for Additional Inspection Control” section on page 20-17
- IP Options—See the “Configuring an IP Options Inspection Policy Map for Additional Inspection Control” section on page 20-21
- MGCP—See the “Configuring an MGCP Inspection Policy Map for Additional Inspection Control” section on page 21-13.
- NetBIOS—See the “Configuring a NetBIOS Inspection Policy Map for Additional Inspection Control” section on page 20-23
- RADIUS Accounting—See the “Configuring a RADIUS Inspection Policy Map for Additional Inspection Control” section on page 23-4
- RTSP—See the “Configuring an RTSP Inspection Policy Map for Additional Inspection Control” section on page 21-16
- SIP—See the “Configuring a SIP Inspection Policy Map for Additional Inspection Control” section on page 21-20
- Skinny—See the “Configuring a Skinny (SCCP) Inspection Policy Map for Additional Inspection Control” section on page 21-26
- SNMP—See the “Configuring an SNMP Inspection Policy Map for Additional Inspection Control” section on page 23-5.

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
```

The default policy map is called “global_policy.” This policy map includes the default inspections listed in the “Default Settings” section on page 19-3. 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
```
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>
<thead>
<tr>
<th>Keywords</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ctiqbe</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 23-2, identify the map name in this command.</td>
</tr>
<tr>
<td>dns [map_name]</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 20-7, 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.</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 20-26, identify the map name in this command.</td>
</tr>
<tr>
<td>ftp [strict [map_name]]</td>
<td>Use the <code>strict</code> keyword to increase the security of protected networks by preventing web browsers from sending embedded commands in FTP requests. See the “Using the <code>strict</code> Option” section on page 20-11 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 20-12, 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 21-6, 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 21-6, identify the map name in this command.</td>
</tr>
<tr>
<td>http</td>
<td>—</td>
</tr>
</tbody>
</table>
### Table 19-2  Protocol Keywords

<table>
<thead>
<tr>
<th>Keywords</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>icmp</td>
<td>—</td>
</tr>
<tr>
<td>icmp error</td>
<td>—</td>
</tr>
<tr>
<td>ils</td>
<td>—</td>
</tr>
<tr>
<td>im [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 20-17, identify the map name in this command.</td>
</tr>
<tr>
<td>ip-options [map_name]</td>
<td>If you added an IP Options inspection policy map according to “Configuring an IP Options Inspection Policy Map for Additional Inspection Control” section on page 20-21, identify the map name in this command.</td>
</tr>
<tr>
<td>mgcp [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 21-13, identify the map name in this command.</td>
</tr>
<tr>
<td>netbios [map_name]</td>
<td>If you added a NetBIOS inspection policy map according to “Configuring a NetBIOS Inspection Policy Map for Additional Inspection Control” section on page 20-23, identify the map name in this command.</td>
</tr>
<tr>
<td>pptp</td>
<td>—</td>
</tr>
<tr>
<td>radius-accounting [map_name]</td>
<td>The radius-accounting keyword is only available for a management class map. See the “Creating a Layer 3/4 Class Map for Management Traffic” section on page 14-13 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 23-4, identify the map name in this command.</td>
</tr>
<tr>
<td>rsh</td>
<td>—</td>
</tr>
<tr>
<td>rtsp [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 21-16, identify the map name in this command.</td>
</tr>
<tr>
<td>sip [map_name]</td>
<td>If you added a SIP inspection policy map according to “Configuring a SIP Inspection Policy Map for Additional Inspection Control” section on page 21-20, identify the map name in this command.</td>
</tr>
<tr>
<td>skinny [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 21-26, 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 19-2 Protocol Keywords

<table>
<thead>
<tr>
<th>Keywords</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>snmp [map_name]</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 23-5, identify the map name in this command.</td>
</tr>
<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 <strong>inspect sunrpc</strong> 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 ASA 1000V to do 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 ASA 1000V by default, but you might need to enable others depending on your network.

This chapter includes the following sections:

- DNS Inspection, page 20-1
- FTP Inspection, page 20-11
- HTTP Inspection, page 20-16
- ICMP Inspection, page 20-16
- ICMP Error Inspection, page 20-16
- Instant Messaging Inspection, page 20-17
- IP Options Inspection, page 20-20
- IPsec Pass Through Inspection, page 20-22
- NetBIOS Inspection, page 20-23
- PPTP Inspection, page 20-25
- SMTP and Extended SMTP Inspection, page 20-25
- TFTP Inspection, page 20-28

DNS Inspection

This section describes DNS application inspection. This section includes the following topics:

- How DNS Application Inspection Works, page 20-2
- How DNS Rewrite Works, page 20-2
- Configuring DNS Rewrite, page 20-3
- Configuring a DNS Inspection Policy Map for Additional Inspection Control, page 20-7
How DNS Application Inspection Works

The ASA 1000V tears down the DNS session associated with a DNS query as soon as the DNS reply is forwarded by the ASA 1000V. The ASA 1000V 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 ASA 1000V 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 ASA 1000V performs reassembly as needed to verify that the packet length is less than the maximum length configured. The ASA 1000V 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 ASA 1000V 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 20-3.

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 20-1, 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 security profile 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 ASA 1000V 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 ASA 1000V 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 20-4.

![Figure 20-1 Translating the Address in a DNS Reply (DNS Rewrite)](image)

DNS rewrite also works if the client making the DNS request is on a DMZ network and the DNS server is on a security profile interface. For an illustration and configuration instructions for this scenario, see the “Overview of DNS Rewrite with Three NAT Zones” section on page 20-4.

**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 20-4
- Overview of DNS Rewrite with Three NAT Zones, page 20-4
- Configuring DNS Rewrite with Three NAT Zones, page 20-6
Configuring DNS Rewrite with Two NAT Zones

To implement a DNS Rewrite scenario similar to the one shown in Figure 20-1, perform the following steps:

**Step 1** Create a static translation for the web server using the `dns` option. See Chapter 12, “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.

```bash
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:

```bash
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 20-7.

**Step 5** On the public DNS server, add an A-record for the web server, such as:

```text
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 ASA 1000V for the scenario shown in Figure 20-1. It assumes DNS inspection is already enabled.

```bash
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:

```text
server.example.com. IN A 209.165.200.225
```

Overview of DNS Rewrite with Three NAT Zones

Figure 20-2 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 20-6.
Figure 20-2  DNS Rewrite with Three NAT Zones

In Figure 20-2, a web server, server.example.com, has the real address 192.168.100.10 on the DMZ interface of the ASA 1000V. A web client with the IP address 10.10.10.25 is on the security profile 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 ASA 1000V at the outside interface.
5. The static rule translates the address 209.165.200.225 to 192.168.100.10 and the ASA 1000V 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 ASA 1000V 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

   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 ASA 1000V 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 20-2, perform the following steps:

**Step 1** Create a static translation for the web server on the DMZ network using the `dns` option. See Chapter 12, “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 20-7.

**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 ASA 1000V for the scenario shown in Figure 20-2. It assumes DNS inspection is already enabled.

```plaintext
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. (Optional) Add one or more regular expressions for use in traffic matching commands according to the “Creating a Regular Expression” section on page 8-11. See the types of text you can match in the **match** commands described in Step 3.
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 8-13.
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.
DNS Inspection

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:

```plaintext
hostname(config)# class-map type inspect dns [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:

```plaintext
hostname(config-cmap)# description string
```

c. (Optional) To match a specific flag that is set in the DNS header, enter the following command:

```plaintext
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:

```plaintext
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:

```plaintext
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:

```plaintext
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:

```plaintext
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:

```plaintext
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 15-2.

**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:
DNS Inspection

hostname(config-pmap-p)# \textbf{tsig enforced action} \{\textbf{drop [log]} \} \{\textbf{[log]}\}

Where the \textbf{count string} argument specifies the maximum number of mismatch instances before a system message log is sent. The \textbf{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# \textbf{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 ASA 1000V within a limited period of time and there is no resource build-up. However, when you enter the \textbf{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 \textbf{show service-policy} command. The following is sample output from the \textbf{show service-policy} command:

hostname# \textbf{show service-policy}
Interface outside:
FTP Inspection

This section describes the FTP inspection engine. This section includes the following topics:

- FTP Inspection Overview, page 20-11
- Using the strict Option, page 20-11
- Configuring an FTP Inspection Policy Map for Additional Inspection Control, page 20-12
- Verifying and Monitoring FTP Inspection, page 20-16

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 ASA 1000V, create an FTP map according to the “Configuring an FTP Inspection Policy Map for Additional Inspection Control” section on page 20-12.

After you enable the `strict` option on an interface, FTP inspection enforces the following behavior:

- An FTP command must be acknowledged before the ASA 1000V allows a new command.
- The ASA 1000V drops connections that send embedded commands.
- The 227 and PORT commands are checked to ensure they do not appear in an error string.
FTP Inspection

Caution

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 xxxxx a1, a2, a3, a4, p1, p2.”
- **TCP stream editing**—The ASA 1000V 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 ASA 1000V 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 8-11. 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 8-13.

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:

```bash
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:

```bash
hostname(config-cmap)# description string
```

c. (Optional) To match a filename for FTP transfer, enter the following command:

```bash
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:

```bash
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:

```bash
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 20-1 for a list of the FTP commands that you can restrict.
FTP Inspection

Chapter 20      Configuring Inspection of Basic Internet Protocols

FTP Inspection

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 `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.

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 `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.

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 `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 20-1 FTP Map request-command deny Options

<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>
FTP Inspection

- Specify the FTP class map that you created in Step 3 by entering the following command:

```
hostname(config-pmap-c)# class class_map_name
```

- 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 15-2.

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-c)# parameters
``` 

b. To mask the greeting banner from the FTP server, enter the following command:

```
hostname(config-pmap-c)# mask-banner
``` 

c. To mask the reply to syst command, enter the following command:

```
hostname(config-pmap-c)# 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-c)# parameters
hostname(config-pmap-c)# mask-banner

hostname(config)# class-map match-all ftp-traffic
hostname(config-cmap)# match port tcp eq ftp

hostname(config)# policy-map ftp-policy
```
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

The HTTP inspection engine enables the system message 304001 when an inside user issues an HTTP GET request:

%ASA-5-304001: user source_address Accessed [JAVA] URL dest_address: url.

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 ASA 1000V 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 ASA 1000V creates translation sessions for intermediate hops that send ICMP error messages, based on the NAT configuration. The ASA 1000V overwrites the packet with the translated IP addresses.

When disabled, the ASA 1000V 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 ASA 1000V reach the outside host without consuming any additional NAT resource. This is
Chapter 20      Configuring Inspection of Basic Internet Protocols

Instant Messaging Inspection

This section describes the IM inspection engine. This section includes the following topics:

- IM Inspection Overview, page 20-17
- Configuring an Instant Messaging Inspection Policy Map for Additional Inspection Control, page 20-17

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 8-11. 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 8-13.

Step 3 (Optional) Create an IM inspection class map by performing the following steps.
Instant Messaging Inspection

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 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 \textit{ip\_address} and the \textit{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:

\begin{verbatim}
hostname(config-cmap)# match [not] version regex (class class_name | regex_name)
\end{verbatim}

Where the \texttt{regex} \texttt{regex\_name} argument is the regular expression you created in Step 1. The \texttt{class} \texttt{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:

\begin{verbatim}
hostname(config-cmap)# match [not] filename regex (class class_name | regex_name)
\end{verbatim}

Where the \texttt{regex} \texttt{regex\_name} argument is the regular expression you created in Step 1. The \texttt{class} \texttt{regex\_class\_name} is the regular expression class map you created in Step 2.

\textbf{Note} Not supported using the MSN IM protocol.

\section*{Step 4}
Create an IM inspection policy map, enter the following command:

\begin{verbatim}
hostname(config)# policy-map type inspect im policy\_map\_name
hostname(config-pmap)#
\end{verbatim}

Where the \texttt{policy\_map\_name} is the name of the policy map. The CLI enters policy-map configuration mode.

\section*{Step 5}
(Optional) To add a description to the policy map, enter the following command:

\begin{verbatim}
hostname(config-pmap)# description string
\end{verbatim}

\section*{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:

  \begin{verbatim}
  hostname(config-pmap)# class class_map_name
  hostname(config-pmap-c)#
  \end{verbatim}

- Specify traffic directly in the policy map using one of the \texttt{match} commands described in Step 3. If you use a \texttt{match not} command, then any traffic that does not match the criterion in the \texttt{match not} command has the action applied.

You can specify multiple \texttt{class} or \texttt{match} commands in the policy map. For information about the order of \texttt{class} and \texttt{match} commands, see the “Defining Actions in an Inspection Policy Map” section on page 15-2.

\section*{Step 7}
Specify the action you want to perform on the matching traffic by entering the following command:

\begin{verbatim}
hostname(config-pmap-c)# \{drop-connection | reset | log\}
\end{verbatim}

Where the \texttt{drop-connection} action closes the connection. The \texttt{reset} action closes the connection and sends a TCP reset to the client. The \texttt{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.

\begin{verbatim}
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"
\end{verbatim}
IP Options Inspection

This section describes the IP Options inspection engine. This section includes the following topics:

- **IP Options Inspection Overview**, page 20-20
- **Configuring an IP Options Inspection Policy Map for Additional Inspection Control**, page 20-21

### 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 ASA 1000V. Configuring this inspection instructs the ASA 1000V 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:

- IP Options Inspection
- Configuring an IP Options Inspection Policy Map for Additional Inspection Control
• 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 ASA 1000V allows RSVP traffic that contains packets with the Router Alert option (option 20).

Dropping RSVP packets containing the Router Alert option can cause problems in VoIP implementations.

When you configure the ASA 1000V 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 ASA 1000V is configured to allow these options, the ASA 1000V 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:
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 20-22
- “Example for Defining an IPsec Pass Through Parameter Map” section on page 20-23

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
```

---

**NetBIOS Inspection**

This section describes the IM inspection engine. This section includes the following topics:

- NetBIOS Inspection Overview, page 20-23
- Configuring a NetBIOS Inspection Policy Map for Additional Inspection Control, page 20-23

**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 ASA 1000V 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 8-11. 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 8-13.

**Step 3**  
Create a NetBIOS inspection policy map, enter the following command:

```
hostname(config)# policy-map type inspect netbios 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 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 15-2.

**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 ASA 1000V 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 20-25
- Configuring an ESMTP Inspection Policy Map for Additional Inspection Control, page 20-26

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 ASA 1000V and by adding monitoring capabilities.

ESMTP is an enhancement to the SMTP protocol and is similar in 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
SMTP and Extended SMTP Inspection

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, ESMTP, HELP, SAML, SEND, SOML, STARTTLS, and VRFY. Along with the support for seven RFC 821 commands (DATA, HELO, MAIL, NOOP, QUIT, RCPT, RSET), the ASA 1000V supports a total of fifteen SMTP commands.

Other extended SMTP commands, such as ATRN, ONEX, VERB, CHUNKING, and private extensions 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><LF>).
- 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 ASA 1000V 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 packet, 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 8-11. 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 8-13.

**Step 3** Create an ESMTP inspection policy map, enter the following command:

```text
hostname(config)# policy-map type inspect esmtp 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:

```text
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:

     ```text
     hostname(config-pmap-c)# 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:

   ```text
   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 15-2.

**Step 6** To configure parameters that affect the inspection engine, perform the following steps:

a. To enter parameters configuration mode, enter the following command:

   ```text
   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 syslog 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 ASA 1000V 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.
Configuring Inspection for Voice and Video 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 ASA 1000V to do 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 ASA 1000V by default, but you might need to enable others depending on your network.

This chapter includes the following sections:

- CTIQBE Inspection, page 21-1
- H.323 Inspection, page 21-3
- MGCP Inspection, page 21-11
- RTSP Inspection, page 21-15
- SIP Inspection, page 21-19
- Skinny (SCCP) Inspection, page 21-25

CTIQBE Inspection

This section describes CTIQBE application inspection. This section includes the following topics:

- CTIQBE Inspection Overview, page 21-1
- Limitations and Restrictions, page 21-2
- Verifying and Monitoring CTIQBE Inspection, page 21-2

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 ASA 1000V.

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 ASA 1000V, 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 ASA 1000V, 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 ASA 1000V. 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 ASA 1000V. 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 ASA 1000V 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
1 in use, 10 most used
hostname# show conn state ctiqbe detail
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 - inside acknowledged FIN, S - awaiting inside SYN,
       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 21-4
- How H.323 Works, page 21-4
- Limitations and Restrictions, page 21-5
- Configuring an H.323 Inspection Policy Map for Additional Inspection Control, page 21-6
- Configuring H.323 and H.225 Timeout Values, page 21-9
- Verifying and Monitoring H.323 Inspection, page 21-9
**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 ASA 1000V supports H.323 through Version 6, including H.323 v3 feature Multiple Calls on One Call Signaling Channel.

With H.323 inspection enabled, the ASA 1000V 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 ASA 1000V.

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 ASA 1000V 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 ASA 1000V dynamically allocates the H.245 connection based on the inspection of the H.225 messages.

*Note*

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 ASA 1000V opens an H.225 connection based on inspection of the ACF and RCF messages.
Chapter 21 Configuring Inspection for Voice and Video Protocols

H.323 Inspection

After inspecting the H.225 messages, the ASA 1000V opens the H.245 channel and then inspects traffic sent over the H.245 channel as well. All H.245 messages passing through the ASA 1000V 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 ASA 1000V must remember the TPKT length to process and decode the messages properly. For each connection, the ASA 1000V keeps a record that contains the TPKT length for the next expected message.

If the ASA 1000V needs to perform NA T 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 ASA 1000V proxy ACKs that TPKT and appends a new TPKT to the H.245 message with the new length.

**Note**
The ASA 1000V 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 ASA 1000V 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 ASA 1000V 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 21-6.

H.239 Support in H.245 Messages

The ASA 1000V 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 ASA 1000V 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 ASA 1000V 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 preformed by ASN.1 coder.

Limitations and Restrictions

The following are some of the known issues and limitations when using H.323 application inspection:
H.323 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 ASA 1000V.
- 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:

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 8-11. 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 8-13.

3. **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
   hostname(config-cmap)#
   ```

   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 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}
   ```
Chapter 21 Configuring Inspection for Voice and Video Protocols

H.323 Inspection

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-p)#
  ```

- 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 15-2.

**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)#
```
Chapter 21      Configuring Inspection for Voice and Video Protocols

H.323 Inspection

b. To enable call setup between H.323 endpoints, enter the following command:

```
hostname(config)# ras-rcf-pinholes enable
```

You can enable call setup between H.323 endpoints when the Gatekeeper is inside the network. The ASA 1000V 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 ASA 1000V 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:

```
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:

```
hostname(config-pmap-p)# call-party-number
```

e. To enforce H.245 tunnel blocking, enter the following command:

```
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:

```
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:

```
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:

```
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 ASA 1000V. 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:

```
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:

```
hostname(config-pmap-p)# state-checking (h225 | ras)
```

The following example shows how to configure phone number filtering:

```
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 21-9
- Monitoring H.245 Sessions, page 21-10
- Monitoring H.323 RAS Sessions, page 21-10

### Monitoring H.225 Sessions

The `show h225` command displays information for H.225 sessions established across the ASA 1000V. 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
  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 ASA 1000V 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 ASA 1000V 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 ASA 1000V. 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 TKTP header because the TKPT 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 an RTCP port of 49607.

**Monitoring H.323 RAS Sessions**

The `show h323-ras` command displays information for H.323 RAS sessions established across the ASA 1000V 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 21-11
- Configuring an MGCP Inspection Policy Map for Additional Inspection Control, page 21-13
- Configuring MGCP Timeout Values, page 21-14
- Verifying and Monitoring MGCP Inspection, page 21-14

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 21-1 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 ASA 1000V 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 ASA 1000V 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 ASA 1000V 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` 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
  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 21-15
- Using RealPlayer, page 21-16
- Restrictions and Limitations, page 21-16
- Configuring an RTSP Inspection Policy Map for Additional Inspection Control, page 21-16

### RTSP Inspection Overview

The RTSP inspection engine lets the ASA 1000V 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 ASA 1000V 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 ASA 1000V parses Setup response messages with a status code of 200. If the response message is travelling inbound, the server is outside relative to the ASA 1000V and dynamic channels need to be opened for connections coming inbound from the server. If the response message is outbound, then the ASA 1000V does not need to open new dynamic channels.

Because RFC 2326 does not require that the client and server ports must be in the SETUP response message, the ASA 1000V keeps state and remembers 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 ASA 1000V 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 ASA 1000V, 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 ASA 1000V, 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 ASA 1000V, add an inspect rtsp port command.

Restrictions and Limitations

The following restrictions apply to the RSTP inspection.

- The ASA 1000V does not support multicast RTSP or RTSP messages over UDP.
- The ASA 1000V does not have the ability to recognize HTTP cloaking where RTSP messages are hidden in the HTTP messages.
- The ASA 1000V 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 ASA 1000V cannot perform NAT on fragmented packets.
- With Cisco IP/TV, the number of translates the ASA 1000V 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 8-10. 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 8-13.

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 \texttt{match not} command. For example, if the \texttt{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 \texttt{match} command, you should identify the traffic directly in the policy map.

\begin{itemize}
\item[a.] Create the class map by entering the following command:
\begin{verbatim}
hostname(config)# class-map type inspect rtsp [match-all | match-any] class_map_name
hostname(config-cmap)#
\end{verbatim}
Where \texttt{class_map_name} is the name of the class map. The \texttt{match-all} keyword is the default, and specifies that traffic must match all criteria to match the class map. The \texttt{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 \texttt{match} commands.

\item[b.] (Optional) To add a description to the class map, enter the following command:
\begin{verbatim}
hostname(config-cmap)# description string
\end{verbatim}

\item[c.] (Optional) To match an RTSP request method, enter the following command:
\begin{verbatim}
hostname(config-cmap)# match [not] request-method method
\end{verbatim}
Where \texttt{method} is the type of method to match (announce, describe, get_parameter, options, pause, play, record, redirect, setup, set_parameter, teardown).

\item[d.] (Optional) To match URL filtering, enter the following command:
\begin{verbatim}
hostname(config-cmap)# match [not] url-filter regex (class class_name | regex_name)
\end{verbatim}
Where the \texttt{regex regex_name} argument is the regular expression you created in Step 1. The \texttt{class regex_class_name} is the regular expression class map you created in Step 2.

\end{itemize}

\begin{itemize}
\item[Step 4] To create an RTSP inspection policy map, enter the following command:
\begin{verbatim}
hostname(config)# policy-map type inspect rtsp policy_map_name
hostname(config-pmap)#
\end{verbatim}
Where the \texttt{policy_map_name} is the name of the policy map. The CLI enters policy-map configuration mode.

\item[Step 5] (Optional) To add a description to the policy map, enter the following command:
\begin{verbatim}
hostname(config-pmap)# description string
\end{verbatim}

\item[Step 6] To apply actions to matching traffic, perform the following steps.
\begin{itemize}
\item[a.] Specify the traffic on which you want to perform actions using one of the following methods:
\begin{itemize}
\item Specify the RTSP class map that you created in Step 3 by entering the following command:
\begin{verbatim}
hostname(config-pmap)# class class_map_name
hostname(config-pmap-c)#
\end{verbatim}
\item Specify traffic directly in the policy map using one of the \texttt{match} commands described in Step 3. If you use a \texttt{match not} command, then any traffic that does not match the criterion in the \texttt{match not} command has the action applied.
\end{itemize}

\item[b.] Specify the action you want to perform on the matching traffic by entering the following command:
\begin{verbatim}
hostname(config-pmap-c)# [{(drop [send-protocol-error] | drop-connection [send-protocol-error] | mask | reset) [log] | rate-limit message_rate]}
\end{verbatim}
\end{itemize}

\end{itemize}
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 15-2.

**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-p)# parameters
```

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 21-19
- SIP Instant Messaging, page 21-19
- Configuring a SIP Inspection Policy Map for Additional Inspection Control, page 21-20
- Configuring SIP Timeout Values, page 21-24
- Verifying and Monitoring SIP Inspection, page 21-24

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 ASA 1000V 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 ASA 1000V, 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 ASA 1000V, 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 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
- Session Initiation Protocol (SIP) Extension for Instant Messaging, RFC 3428
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 ASA 1000V 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 ASA 1000V, unless the ASA 1000V 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 8-10. 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 8-13.
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.

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 regex_class_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 regex_class_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 regex_class_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}
```
Chapter 21 Configuring Inspection for Voice and Video Protocols

SIP Inspection

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 a 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 15-2.

**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 ASA 1000V. 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-2e43-228f@10.130.56.44
        state Call init, idle 0:00:01
    call-id c3943000-860ca-7ef@11f@10.130.56.45
        state Active, idle 0:00:06
This sample shows two active SIP sessions on the ASA 1000V (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 21-25
- Supporting Cisco IP Phones, page 21-25
- Restrictions and Limitations, page 21-26
- Configuring a Skinny (SCCP) Inspection Policy Map for Additional Inspection Control, page 21-26
- Verifying and Monitoring SIP Inspection, page 21-24

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 ASA 1000V 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 ASA 1000V.

Normal traffic between Cisco CallManager and Cisco IP Phones uses SCCP and is handled by SCCP inspection without any special configuration. The ASA 1000V 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. The ASA 1000V supports inspection of traffic from Cisco IP Phones running SCCP protocol version 19 and earlier.

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.
Skinny (SCCP) Inspection

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 ASA 1000V currently does not support NAT or PAT for the file content transferred over TFTP. Although the ASA 1000V supports NAT of TFTP messages and opens a pinhole for the TFTP file, the ASA 1000V 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 ASA 1000V 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 8-10. 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 8-13.

Step 3  Create an SCCP inspection policy map, enter the following command:

```
hostname(config)# policy-map type inspect skinny 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 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 15-2.

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)# 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:
   ```
   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
   ```
Skinny (SCCP) Inspection

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 ASA 1000V. 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 ASA 1000V to do 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 ASA 1000V by default, but you might need to enable others depending on your network.

This chapter includes the following sections:

- **ILS Inspection**, page 22-1
- **SQL*Net Inspection**, page 22-2
- **Sun RPC Inspection**, page 22-3

**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 ASA 1000V 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 ASA 1000V 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.
SQL*Net Inspection

SQL*Net inspection is enabled by default.

The SQL*Net protocol consists of different packet types that the ASA 1000V handles to make the data stream appear consistent to the Oracle applications on either side of the ASA 1000V.

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.

Disable SQL*Net inspection when SQL data transfer occurs on the same port as the SQL control TCP port 1521. The ASA 1000V 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 ASA 1000V 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.

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.
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 ASA 1000V, 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 22-3
- Managing Sun RPC Services, page 22-4
- Verifying and Monitoring Sun RPC Inspection, page 22-4

**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 ASA 1000V intercepts this packet and opens both embryonic TCP and UDP connections on that port.

The following limitations apply to Sun RPC inspection:

- NAT or PAT of Sun RPC payload information is not supported.
- Sun RPC inspection supports inbound access lists only. Sun RPC inspection does not support outbound access lists because the inspection engine uses dynamic access lists instead of secondary connections. Dynamic access lists are always added on the ingress direction and not on egress; therefore, this inspection engine does not support outbound access lists. To view the dynamic access lists configured for the ASA 1000V, 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.
Managing Sun RPC Services

Use the Sun RPC services table to control Sun RPC traffic through the ASA 1000V 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 209.165.200.5:808 in 192.168.100.2:2049 idle 0:00:04 flags -
UDP out 209.165.200.5:714 in 192.168.100.2:111 idle 0:00:04 flags -
UDP out 209.165.200.5:712 in 192.168.100.2:647 idle 0:00:05 flags -
UDP out 192.168.100.2:0 in 209.165.200.5:714 idle 0:00:05 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:

```
hosname(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 635 status
100003 2 udp 2049 nfs
100003 3 udp 2049 nfs
100003 3 tcp 2049 nfs
100003 3 tcp 2049 nfs
100003 2 tcp 32771 nlockmgr
100003 3 udp 32771 nlockmgr
100003 4 udp 32771 nlockmgr
100003 1 tcp 32852 nlockmgr
100003 3 tcp 32852 nlockmgr
100003 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 ASA 1000V to do 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 ASA 1000V by default, but you might need to enable others depending on your network.

This chapter includes the following sections:
- DCERPC Inspection, page 23-1
- RADIUS Accounting Inspection, page 23-3
- RSH Inspection, page 23-4
- SNMP Inspection, page 23-4
- XDMCP Inspection, page 23-5

DCERPC Inspection

This section describes the DCERPC inspection engine. This section includes the following topics:
- DCERPC Overview, page 23-1
- Configuring a DCERPC Inspection Policy Map for Additional Inspection Control, page 23-2

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 ASA 1000V 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 ASA 1000V. 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
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 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
```

**RADIUS Accounting Inspection**

This section describes the IM inspection engine. This section includes the following topics:

- RADIUS Accounting Inspection Overview, page 23-3
- Configuring a RADIUS Inspection Policy Map for Additional Inspection Control, page 23-4

**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 ASA 1000V 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 ASA 1000V 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 ASA 1000V can validate the message. If the shared secret is not configured, the ASA 1000V 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 ASA 1000V checks for the 3GPP-Session-Stop-Indicator in the Accounting Request STOP messages to properly handle secondary PDP contexts. Specifically, the ASA 1000V 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 23-4
- Configuring an SNMP Inspection Policy Map for Additional Inspection Control, page 23-5

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 ASA 1000V 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 19-5.
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 ASA 1000V must allow the TCP back connection from the Xhosted computer. To permit the back connection, use the `established` command on the ASA 1000V. 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|n. Each display has a separate connection to the Xserver, as a result of the following terminal setting:

```
setenv DISPLAY Xserver:n
```

where `n` is the display number.

When XDMCP is used, the display is negotiated using IP addresses, which the ASA 1000V can NAT if needed. XDCMP inspection does not support PAT.
Chapter 23  Configuring Inspection for Management Application Protocols

XDMCP Inspection
PART 8

Configuring Connection Settings
CHAPTER 24

Configuring Connection Settings

This chapter describes how to configure connection settings for connections that go through the ASA 1000V, or for management connections, that go to the ASA 1000V. 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 24-1
- If you have asymmetric routing configured on upstream routers, and traffic alternates between two ASA 1000Vs, 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 ASA 1000V, 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., page 24-5
- Guidelines and Limitations, page 24-5
- Default Settings, page 24-5
- Configuring Connection Settings, page 24-6
- Monitoring Connection Settings, page 24-14
- Configuration Examples for Connection Settings, page 24-15
- Feature History for Connection Settings, page 24-16

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 24-2
- Disabling TCP Intercept for Management Packets for Clientless SSL Compatibility, page 24-2
Information About Connection Settings

- Dead Connection Detection (DCD), page 24-2
- TCP Sequence Randomization, page 24-3
- TCP Normalization, page 24-3
- TCP State Bypass, page 24-3

TCP Intercept and Limiting Embryonic Connections

Limiting the number of embryonic connections protects you from a DoS attack. The ASA 1000V 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 ASA 1000V acts as a proxy for the server and generates a SYN-ACK response to the client SYN request. When the ASA 1000V 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 cannot access the server during a SYN attack.

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 ASA 1000V 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 ASA 1000V 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 ASA 1000V, and the eBGP peers are using MD5. Randomization breaks the MD5 checksum.
- You use a WAAS device that requires the ASA 1000V not to randomize the sequence numbers of connections.

**TCP Normalization**

The TCP normalization feature identifies abnormal packets that the ASA 1000V can act on when they are detected; for example, the ASA 1000V can allow, drop, or clear the packets. TCP normalization helps protect the ASA 1000V 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 24-6) 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 ASA 1000V includes SYN flood protection in other ways.
- The normalizer always sees the SYN packet as the first packet in a flow unless the ASA 1000V is in loose mode due to failover.

**TCP State Bypass**

By default, all traffic that goes through the ASA 1000V is inspected using the Adaptive Security Algorithm and is either allowed through or dropped based on the security policy. The ASA 1000V 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 ASA 1000V 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 ASA 1000V.

For example, a new connection goes to ASA 1000V 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 ASA 1000V 1, then the packets will match the entry in the fast path, and are passed through. But if subsequent packets go to ASA 1000V 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 24-1 shows an asymmetric routing example where the outbound traffic goes through a different ASA 1000V than the inbound traffic:

![Asymmetric Routing Diagram](image_url)

If you have asymmetric routing configured on upstream routers, and traffic alternates between two ASA 1000Vs, 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 ASA 1000V, 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.

**Guidelines and Limitations**

**TCP State Bypass Guidelines and Limitations**

**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 ASA 1000V, so application inspection is not supported with TCP state bypass.
- AAA authenticated sessions—When a user authenticates with one ASA 1000V, traffic returning via the other ASA 1000V will be denied because the user did not authenticate with that ASA 1000V.
- TCP Intercept, maximum embryonic connection limit, TCP sequence number randomization—The ASA 1000V 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 ASA 1000V, be sure to configure static NAT on both ASA 1000Vs for TCP state bypass traffic; if you use dynamic NAT, the address chosen for the session on ASA 1000V 1 will differ from the address chosen for the session on ASA 1000V 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`
- `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`
Configuring Connection Settings

This section includes the following topics:
- Customizing the TCP Normalizer with a TCP Map, page 24-6
- Configuring Connection Settings, page 24-10

Task Flow For Configuring Configuration Settings (Except Global Timeouts)

<table>
<thead>
<tr>
<th>Step 1</th>
<th>For TCP normalization customization, create a TCP map according to the “Customizing the TCP Normalizer with a TCP Map” section on page 24-6.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 2</td>
<td>For all connection settings except for global timeouts, configure a service policy according to Chapter 14, “Configuring a Service Policy Using the Modular Policy Framework.”</td>
</tr>
<tr>
<td>Step 3</td>
<td>Configure connection settings according to the “Configuring Connection Settings” section on page 24-10.</td>
</tr>
</tbody>
</table>

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 24-1). If you want to customize some settings, then the defaults are used for any commands you do not enter. |
### Table 24-1  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>
<tr>
<td>exceed-mss {allow</td>
<td>drop}</td>
</tr>
<tr>
<td></td>
<td>(Default) The <strong>allow</strong> keyword allows packets whose data length exceeds the TCP maximum segment size.</td>
</tr>
<tr>
<td></td>
<td>The <strong>drop</strong> keyword drops packets whose data length exceeds the TCP maximum segment size.</td>
</tr>
<tr>
<td>invalid-ack {allow</td>
<td>drop}</td>
</tr>
<tr>
<td></td>
<td>• 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.</td>
</tr>
<tr>
<td></td>
<td>• 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.</td>
</tr>
<tr>
<td></td>
<td>The <strong>allow</strong> keyword allows packets with an invalid ACK.</td>
</tr>
<tr>
<td></td>
<td>(Default) The <strong>drop</strong> keyword drops packets with an invalid ACK.</td>
</tr>
<tr>
<td><strong>Note</strong></td>
<td>TCP packets with an invalid ACK are automatically allowed for WAAS connections.</td>
</tr>
</tbody>
</table>
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 ASA 1000V 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 24-1  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>ttl-evasion-protection</td>
<td>Disables the TTL evasion protection. Do not enter 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 ASA 1000V 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 ASA 1000V 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>
Table 24-1 tcp-map Commands (continued)

<table>
<thead>
<tr>
<th>Command</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>urgent-flag {allow</td>
<td>Sets the action for packets with the URG flag. The URG flag is used to indicate that the packet contains information that is of</td>
</tr>
<tr>
<td>clear}</td>
<td>higher priority than other data within the stream. The TCP RFC is vague about the exact interpretation of the URG flag, therefore end</td>
</tr>
<tr>
<td></td>
<td>systems handle urgent offsets in different ways, which may make the end system vulnerable to attacks.</td>
</tr>
<tr>
<td></td>
<td>The allow keyword allows packets with the URG flag.</td>
</tr>
<tr>
<td></td>
<td>(Default) The clear keyword clears the URG flag and allows the packet.</td>
</tr>
<tr>
<td>window-variation</td>
<td>Sets the action for a connection that has changed its window size unexpectedly. The window size mechanism allows TCP to advertise</td>
</tr>
<tr>
<td>{allow</td>
<td>drop}</td>
</tr>
<tr>
<td></td>
<td>“shrinking the window” is strongly discouraged. When this condition is detected, the connection can be dropped.</td>
</tr>
<tr>
<td></td>
<td>(Default) The allow keyword allows connections with a window variation.</td>
</tr>
<tr>
<td></td>
<td>The drop keyword drops connections with a window variation.</td>
</tr>
</tbody>
</table>

Guidelines and Limitations

Depending on the number of CPU cores on your ASA 1000V 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 ASA 1000V 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.
## Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><strong>class-map name</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><code>hostname(config)# class-map bypass_traffic</code></td>
</tr>
<tr>
<td></td>
<td>Creates a class map to identify the traffic for which you want to</td>
</tr>
<tr>
<td></td>
<td>disable stateful firewall inspection.</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><strong>match parameter</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><code>hostname(config-cmap)# match access-list bypass</code></td>
</tr>
<tr>
<td></td>
<td>Specifies the traffic in the class map. See the “Identifying Traffic</td>
</tr>
<tr>
<td></td>
<td>(Layer 3/4 Class Maps)” section on page 14-10 for more information.</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td><strong>policy-map name</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><code>hostname(config)# policy-map tcp_bypass_policy</code></td>
</tr>
<tr>
<td></td>
<td>Adds or edits a policy map that sets the actions to take with the</td>
</tr>
<tr>
<td></td>
<td>class map traffic.</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td><strong>class name</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><code>hostname(config-pmap)# class bypass_traffic</code></td>
</tr>
<tr>
<td></td>
<td>Identifies the class map created in <strong>Step 1</strong></td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td>Do one or more of the following:</td>
</tr>
</tbody>
</table>
### Configuring Connection Settings

**set connection**

- **conn-max n**
- **embryonic-conn-max n**
- **per-client-embryonic-max n**
- **per-client-max n**
- **random-sequence-number**

#### Example:

```
hostname(config-pmap-c)# set connection
conn-max 256 random-sequence-number
disable
```

#### Command Purpose

Sets maximum connection limits or whether TCP sequence randomization is enabled.

The **conn-max n** argument sets the maximum number of simultaneous TCP and/or UDP connections that are allowed, between 0 and 65535. The default is 0, which allows unlimited connections.

If two servers are configured to allow simultaneous TCP and/or UDP connections, the connection limit is applied to each configured server separately.

When configured under a class, this argument restricts the maximum number of simultaneous connections that are allowed for the entire class. In this case, one attack host can consume all the connections and leave none of the rest of the hosts matched in the access list under the class.

The **embryonic-conn-max n** argument sets the maximum number of simultaneous embryonic connections allowed, between 0 and 65535. The default is 0, which allows unlimited connections.

The **per-client-embryonic-max n** argument sets the maximum number of simultaneous embryonic connections allowed per client, between 0 and 65535. The default is 0, which allows unlimited connections.

The **per-client-max n** argument sets the maximum number of simultaneous connections allowed per client, between 0 and 65535. The default is 0, which allows unlimited connections. When configured under a class, this argument restricts the maximum number of simultaneous connections that are allowed for each host that is matched through an access list under the class.

The **random-sequence-number** **enable** | **disable**keyword enables or disables TCP sequence number randomization. See the “TCP Sequence Randomization” section on page 24-3 section for more information.

You can enter this command all on one line (in any order), or you can enter each attribute as a separate command. The ASA 1000V combines the command into one line in the running configuration.

#### Note

For management traffic, you can only set the **conn-max** and **embryonic-conn-max** keywords.
Chapter 24      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

Sets connection timeouts.

The embryonic hh:mm:ss keyword sets the timeout period until a TCP embryonic (half-open) connection is closed, between 0:0:5 and 1193:00:00. 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 0:5:0 and 1193:00:00. 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 1193:00:00. The default is 0:10:0. Half-closed connections are not affected by DCD. Also, the ASA 1000V 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 ASA 1000V 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 ASA 1000V frees the connection. If both end hosts respond that the connection is valid, the ASA 1000V 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 minutes.

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.
Floating Connection—When multiple static routes exist to a network with different metrics, the ASA 1000V uses the one with the best metric at the time of connection creation. If a better route becomes available, then this timeout lets connections be closed so a connection can be reestablished to use the better route. The default is 0 (the connection never times out). To take advantage of this feature, change the timeout to a new value between 0:1:0 and 1193:0:0.

(8.4(3) and later, not including 8.5(1) and 8.6(1)) PAT Translation Slot—Modifies the idle time until a PAT translation slot is freed, between 0:0:30 and 0:5:0. The default is 30 seconds. You may want to increase the timeout if upstream routers reject new connections using a freed PAT port because the previous connection might still be open on the upstream device.

---

### Monitoring Connection Settings

This section includes the following topics:

- Monitoring TCP State Bypass, page 24-14

---

### 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><code>show conn</code></td>
<td>If you use the <code>show conn</code> 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 24-15
- Configuration Examples for TCP State Bypass, page 24-15
- Configuration Examples for TCP Normalization, page 24-16

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)# 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 ASA 1000V 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 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>TCP state bypass</td>
<td>8.2(1)</td>
<td>This feature was introduced. The following command was introduced:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>set connection advanced-options tcp-state-bypass.</td>
</tr>
<tr>
<td>Connection timeout for all protocols</td>
<td>8.2(2)</td>
<td>The idle timeout was changed to apply to all protocols, not just TCP.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The following command was modified:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>set connection timeout</td>
</tr>
<tr>
<td>Timeout for connections using a backup static</td>
<td>8.2(5)/8.4(2)</td>
<td>When multiple static routes exist to a network with different metrics, the ASA</td>
</tr>
<tr>
<td>route</td>
<td></td>
<td>1000V uses the one with the best metric at the time of connection creation. If a</td>
</tr>
<tr>
<td></td>
<td></td>
<td>better route becomes available, then this timeout lets connections be closed so a</td>
</tr>
<tr>
<td></td>
<td></td>
<td>connection can be reestablished to use the better route. The default is 0 (the</td>
</tr>
<tr>
<td></td>
<td></td>
<td>connection never times out). To take advantage of this feature, change the timeout</td>
</tr>
<tr>
<td></td>
<td></td>
<td>to a new value.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>We modified the following command:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>timeout floating-conn.</td>
</tr>
<tr>
<td>Configurable timeout for PAT xlate</td>
<td>8.4(3)</td>
<td>When a PAT xlate times out (by default after 30 seconds), and the ASA 1000V reuses</td>
</tr>
<tr>
<td></td>
<td></td>
<td>the port for a new translation, some upstream routers might reject the new connection</td>
</tr>
<tr>
<td></td>
<td></td>
<td>because the previous connection might still be open on the upstream device. The</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PAT xlate timeout is now configurable, to a value between 30 seconds and 5 minutes.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>We introduced the following command:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>timeout pat-xlate.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This feature is not available in 8.5(1) or 8.6(1).</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 25-1
- Configuring the Fragment Size, page 25-2
- Blocking Unwanted Connections, page 25-2
- Configuring IP Audit for Basic IPS Support, page 25-3

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 ASA 1000V only looks at the destination address when determining where to forward the packet. Unicast RPF instructs the ASA 1000V 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 ASA 1000V, the ASA 1000V routing table must include a route back to the source address. See RFC 2267 for more information.

For outside traffic, for example, the ASA 1000V 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 ASA 1000V 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 ASA 1000V drops the packet. Similarly, if traffic enters the inside interface from an unknown source address, the ASA 1000V 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 ASA 1000V 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 ASA 1000V. 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, system log messages show an attack), then you can block (or shun) connections based on the source IP address. No new connections can be made 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 ASA 1000V 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.

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 ASA 1000V that does not have an AIP SSM. It supports a basic list of signatures, and you can configure the ASA 1000V to perform one or more actions on traffic that matches a signature.

This section includes the following topics:

- Configuring IP Audit, page 25-3
- IP Audit Signature List, page 25-4

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:

```
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:

```
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:

```
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 25-1 lists supported signatures and system message numbers.

**Table 25-1 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 25-1 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>
<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>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>
</tbody>
</table>
Table 25-1 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>3042</td>
<td>400028</td>
<td>TCP FIN only flags</td>
<td>Attack</td>
<td>Triggers when a single orphaned TCP FIN packet is sent to a privileged port (having port number less than 1024) on a specific host.</td>
</tr>
<tr>
<td>3153</td>
<td>400029</td>
<td>FTP Improper Address Specified</td>
<td>Informational</td>
<td>Triggers if a port command is issued with an address that is not the same as the requesting host.</td>
</tr>
<tr>
<td>3154</td>
<td>400030</td>
<td>FTP Improper Port Specified</td>
<td>Informational</td>
<td>Triggers if a port command is issued with a data port specified that is &lt;1024 or &gt;65535.</td>
</tr>
<tr>
<td>4050</td>
<td>400031</td>
<td>UDP Bomb attack</td>
<td>Attack</td>
<td>Triggers when the UDP length specified is less than the IP length specified. This malformed packet type is associated with a denial of service attempt.</td>
</tr>
<tr>
<td>4051</td>
<td>400032</td>
<td>UDP Snork attack</td>
<td>Attack</td>
<td>Triggers when a UDP packet with a source port of either 135, 7, or 19 and a destination port of 135 is detected.</td>
</tr>
<tr>
<td>4052</td>
<td>400033</td>
<td>UDP Chargen DoS attack</td>
<td>Attack</td>
<td>This signature triggers when a UDP packet is detected with a source port of 7 and a destination port of 19.</td>
</tr>
<tr>
<td>6050</td>
<td>400034</td>
<td>DNS HINFO Request</td>
<td>Informational</td>
<td>Triggers on an attempt to access HINFO records from a DNS server.</td>
</tr>
<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>
</tbody>
</table>
### Table 25-1  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>6152</td>
<td>400044</td>
<td>yp passwd (YP password daemon) Portmap Request</td>
<td>Informational</td>
<td>Triggers when a request is made to the portmapper for the YP password daemon (yp passwd) port.</td>
</tr>
<tr>
<td>6153</td>
<td>400045</td>
<td>yp updated (YP update daemon) Portmap Request</td>
<td>Informational</td>
<td>Triggers when a request is made to the portmapper for the YP update daemon (yp updated) 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 (yp xfrd) 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>
<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>
P A R T  9

Configuring VPN
This chapter describes how to configure Internet Protocol Security (IPsec) and the Internet Security Association and Key Management Protocol (ISAKMP) standards to build Virtual Private Networks (VPNs). It includes the following sections:

- Information About Tunneling, IPsec, and ISAKMP, page 26-1
- Configuring ISAKMP, page 26-3
- Using the Tunnel-group-map default-group Command, page 26-9
- Configuring IPsec, page 26-9
- Clearing Security Associations, page 26-26
- Clearing Crypto Map Configurations, page 26-26

Information About Tunneling, IPsec, and ISAKMP

Tunneling makes it possible to use a public TCP/IP network, such as the Internet, to create secure connections between remote users and a private corporate network. Each secure connection is called a tunnel.

The ASA 1000V uses the ISAKMP and IPsec tunneling standards to build and manage tunnels. ISAKMP and IPsec accomplish the following:

- Negotiate tunnel parameters
- Establish tunnels
- Authenticate users and data
- Manage security keys
- Encrypt and decrypt data
- Manage data transfer across the tunnel
- Manage data transfer inbound and outbound as a tunnel endpoint or router

The ASA 1000V functions as a bidirectional tunnel endpoint. It can receive plain packets from the private network, encapsulate them, create a tunnel, and send them to the other end of the tunnel where they are unencapsulated and sent to their final destination. It can also receive encapsulated packets from the public network, unencapsulate them, and send them to their final destination on the private network.
IPsec Overview

The ASA 1000V uses IPsec for LAN-to-LAN VPN connections. In IPsec terminology, a peer is another secure gateway. The ASA 1000V supports only Cisco peers. Because we adhere to VPN industry standards, ASAs can work with other vendors’ peers; however, we do not support them.

During tunnel establishment, the two peers negotiate security associations that govern authentication, encryption, encapsulation, and key management. These negotiations involve two phases: first, to establish the tunnel (the IKE SA) and second, to govern traffic within the tunnel (the IPsec SA).

A LAN-to-LAN VPN connects networks in different geographic locations. In IPsec LAN-to-LAN connections, the ASA 1000V can function as initiator or responder. Initiators propose SAs; responders accept, reject, or make counter-proposals—all in accordance with configured SA parameters. To establish a connection, both entities must agree on the SAs.

Note
When the ASA 1000V is configured for IPsec VPN, you cannot enable security contexts (also called firewall multimode) or Active/Active stateful failover. Therefore, these features are unavailable.

ISAKMP and IKE Overview

ISAKMP is the negotiation protocol that lets two hosts agree on how to build an IPsec security association (SA). It provides a common framework for agreeing on the format of SA attributes. This security association includes negotiating with the peer about the SA and modifying or deleting the SA. ISAKMP separates negotiation into two phases: Phase 1 and Phase 2. Phase 1 creates the first tunnel, which protects later ISAKMP negotiation messages. Phase 2 creates the tunnel that protects data.

IKE uses ISAKMP to set up the SA for IPsec to use. IKE creates the cryptographic keys used to authenticate peers.

To set the terms of the ISAKMP negotiations, you create an IKE policy, which includes the following:

- The authentication type required of the IKEv1 peer, either RSA signature using certificates or preshared key (PSK).
- An encryption method to protect the data and ensure privacy.
- A Hashed Message Authentication Codes (HMAC) method to ensure the identity of the sender, and to ensure that the message has not been modified in transit.
- A Diffie-Hellman group to determine the strength of the encryption-key-determination algorithm. The ASA 1000V uses this algorithm to derive the encryption and hash keys.
- For IKEv2, a separate pseudo-random function (PRF) used as the algorithm to derive keying material and hashing operations required for the IKEv2 tunnel encryption and so on.
- A limit to the time the ASA 1000V uses an encryption key before replacing it.

With IKEv1 policies, you set one value for each parameter. For IKEv2, you can configure multiple encryption and authentication types, and multiple integrity algorithms for a single policy. The ASA 1000V orders the settings from the most secure to the least secure and negotiates with the peer using that order. This ordering allows you to potentially send a single proposal to convey all the allowed transforms instead of sending each allowed combination as with IKEv1.
Configuring ISAKMP

This section describes the Internet Security Association and Key Management Protocol (ISAKMP) and the Internet Key Exchange (IKE) protocol.

This section includes the following topics:
- Configuring IKEv1 and IKEv2 Policies, page 26-3
- Enabling IKE on the Outside Interface, page 26-7
- Disabling IKEv1 Aggressive Mode, page 26-7
- Determining an ID Method for IKEv1 and IKEv2 ISAKMP Peers, page 26-7
- Enabling IPsec over NAT-T, page 26-8
- Waiting for Active Sessions to Terminate Before Rebooting, page 26-9
- Alerting Peers Before Disconnecting, page 26-9

Configuring IKEv1 and IKEv2 Policies

To create an IKE policy, enter the `crypto ikev1 | ikev2 policy` command from global configuration mode. The prompt displays IKE policy configuration mode. For example:

```
hostname(config)# crypto ikev1 policy 1
hostname(config-ikev1-policy)#
```

After creating the policy, you can specify the settings for the policy.

Table 26-1 and Table 26-2 provide information about the IKEv1 and IKEv2 policy keywords and their values.

### Table 26-1 IKEv1 Policy Keywords for CLI Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Keyword</th>
<th>Meaning</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>authentication</td>
<td>rsa-sig</td>
<td>A digital certificate with keys generated by the RSA signatures algorithm</td>
<td>Specifies the authentication method the ASA 1000V uses to establish the identity of each IPsec peer.</td>
</tr>
<tr>
<td>crack</td>
<td>Challenge/Response for Authenticated Cryptographic Keys</td>
<td>CRACK provides strong mutual authentication when the client authenticates using a legacy method such as RADIUS, and the server uses public key authentication.</td>
<td></td>
</tr>
<tr>
<td>pre-share (default)</td>
<td>Preshared keys</td>
<td>Preshared keys do not scale well with a growing network but are easier to set up in a small network.</td>
<td></td>
</tr>
<tr>
<td>encryption</td>
<td>des</td>
<td>56-bit DES-CBC</td>
<td>Specifies the symmetric encryption algorithm that protects data transmitted between two IPsec peers. The default is 168-bit Triple DES.</td>
</tr>
<tr>
<td>3des (default)</td>
<td>168-bit Triple DES</td>
<td>The Advanced Encryption Standard supports key lengths of 128, 192, 256 bits.</td>
<td></td>
</tr>
<tr>
<td>aes</td>
<td>aes-192</td>
<td></td>
<td></td>
</tr>
<tr>
<td>aes-256</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Cisco ASA 1000V CLI Configuration Guide for ASDM Mode
Chapter 26      Configuring IPsec and ISAKMP

Configuring ISAKMP

Table 26-1  IKEv1 Policy Keywords for CLI Commands (continued)

<table>
<thead>
<tr>
<th>Command</th>
<th>Keyword</th>
<th>Meaning</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>hash</td>
<td>sha (default)</td>
<td>SHA-1 (HMAC variant)</td>
<td>Specifies the hash algorithm used to ensure data integrity. It ensures that a packet comes from where it says it comes from and that it has not been modified in transit.</td>
</tr>
<tr>
<td>md5</td>
<td>MD5 (HMAC variant)</td>
<td></td>
<td>The default is SHA-1. MD5 has a smaller digest and is considered to be slightly faster than SHA-1. A successful (but extremely difficult) attack against MD5 has occurred; however, the HMAC variant IKE uses prevents this attack.</td>
</tr>
<tr>
<td>group</td>
<td>1</td>
<td>Group 1 (768-bit)</td>
<td>Specifies the Diffie-Hellman group identifier, which the two IPsec peers use to derive a shared secret without transmitting it to each other.</td>
</tr>
<tr>
<td></td>
<td>2 (default)</td>
<td>Group 2 (1024-bit)</td>
<td>The lower the Diffie-Hellman group number, the less CPU time it requires to execute. The higher the Diffie-Hellman group number, the greater the security. AES support is available on security appliances licensed for VPN-3DES only. To support the large key sizes required by AES, ISAKMP negotiation should use Diffie-Hellman (DH) Group 5.</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>Group 5 (1536-bit)</td>
<td></td>
</tr>
<tr>
<td>lifetime</td>
<td>integer value</td>
<td>120 to 2147483647 seconds</td>
<td>Specifies the SA lifetime. The default is 86,400 seconds or 24 hours. As a general rule, a shorter lifetime provides more secure ISAKMP negotiations (up to a point). However, with shorter lifetimes, the ASA 1000V sets up future IPsec SAs more quickly.</td>
</tr>
</tbody>
</table>

Table 26-2  IKEv2 Policy Keywords for CLI Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Keyword</th>
<th>Meaning</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>integrity</td>
<td>sha (default)</td>
<td>SHA-1 (HMAC variant)</td>
<td>Specifies the hash algorithm used to ensure data integrity. It ensures that a packet comes from where it says it comes from and that it has not been modified in transit.</td>
</tr>
<tr>
<td></td>
<td>md5</td>
<td>MD5 (HMAC variant)</td>
<td>The default is SHA-1. MD5 has a smaller digest and is considered to be slightly faster than SHA-1. A successful (but extremely difficult) attack against MD5 has occurred; however, the HMAC variant IKE user prevents this attack.</td>
</tr>
<tr>
<td>sha256</td>
<td>SHA 2, 256-bit digest</td>
<td></td>
<td>Specifies the Secure Hash Algorithm SHA 2 with the 256-bit digest.</td>
</tr>
<tr>
<td>sha384</td>
<td>SHA 2, 384-bit digest</td>
<td></td>
<td>Specifies the Secure Hash Algorithm SHA 2 with the 384-bit digest.</td>
</tr>
<tr>
<td>sha512</td>
<td>SHA 2, 512-bit digest</td>
<td></td>
<td>Specifies the Secure Hash Algorithm SHA 2 with the 512-bit digest.</td>
</tr>
</tbody>
</table>
IKEv1 and IKEv2 each support a maximum of 20 IKE policies, each with a different set of values. Assign a unique priority to each policy that you create. The lower the priority number, the higher the priority.

When IKE negotiations begin, the peer that initiates the negotiation sends all of its policies to the remote peer, and the remote peer tries to find a match. The remote peer checks all of the peer’s policies against each of its configured policies in priority order (highest priority first) until it discovers a match.

A match exists when both policies from the two peers contain the same encryption, hash, authentication, and Diffie-Hellman parameter values. For IKEv1, the remote peer policy must also specify a lifetime less than or equal to the lifetime in the policy the initiator sent. If the lifetimes are not identical, the
ASA 1000V uses the shorter lifetime. For IKEv2 the lifetime is not negotiated but managed locally between each peer, making it possible to configure lifetime independently on each peer. If no acceptable match exists, IKE refuses negotiation and the SA is not established.

There is an implicit trade-off between security and performance when you choose a specific value for each parameter. The level of security the default values provide is adequate for the security requirements of most organizations. If you are interoperating with a peer that supports only one of the values for a parameter, your choice is limited to that value.

**Note**

New ASA configurations do not have a default IKEv1 or IKEv2 policy.

To configure IKE policies, in global configuration mode, use the `crypto ikev1 | ikev2 policy` command to enter IKE policy configuration mode:

```
crypto ikev1 | ikev2 policy priority
```

You must include the priority in each of the ISAKMP commands. The priority number uniquely identifies the policy and determines the priority of the policy in IKE negotiations.

To enable and configure IKE, complete the following steps, using the IKEv1 examples as a guide:

**Note**

If you do not specify a value for a given policy parameter, the default value applies.

---

**Step 1**

Enter IKEv1 policy configuration mode:

```
hostname(config)# crypto ikev1 policy 1
hostname(config-ikev1-policy)#
```

**Step 2**

Specify the encryption algorithm. The default is Triple DES. This example sets encryption to DES.

```
encryption [aes | aes-192 | aes-256 | des | 3des]
```

For example:

```
hostname(config-ikev1-policy)# encryption des
```

**Step 3**

Specify the hash algorithm. The default is SHA-1. This example configures MD5.

```
hash [md5 | sha]
```

For example:

```
hostname(config-ikev1-policy)# hash md5
```

**Step 4**

Specify the authentication method. The default is preshared keys. This example configures RSA signatures.

```
authentication [pre-share | crack | rsa-sig]
```

For example:

```
hostname(config-ikev1-policy)# authentication rsa-sig
```

**Step 5**

Specify the Diffie-Hellman group identifier. The default is Group 2. This example configures Group 5.

```
group [1 | 2 | 5]
```

For example:

```
hostname(config-ikev1-policy)# group 5
```
Step 6  Specify the SA lifetime. This examples sets a lifetime of 4 hours (14400 seconds). The default is 86400 seconds (24 hours).

```console
lifetime seconds
```
For example:
```console
hostname(config-ikev1-policy)# lifetime 14400
```

---

### Enabling IKE on the Outside Interface

You must enable IKE on the interface that terminates the VPN tunnel. Typically this is the outside, or public interface. To enable IKEv1 or IKEv2, use the `crypto ikev1 | ikev2 enable` command from global configuration mode:

```console
crypto ikev1 | ikev2 enable interface-name
```
For example:
```console
hostname(config)# crypto ikev1 enable outside
```

---

### Disabling IKEv1 Aggressive Mode

Phase 1 IKEv1 negotiations can use either main mode or aggressive mode. Both provide the same services, but aggressive mode requires only two exchanges between the peers totaling three messages, rather than three exchanges totaling six messages. Aggressive mode is faster, but does not provide identity protection for the communicating parties. Therefore, the peers must exchange identification information before establishing a secure SA. Aggressive mode is enabled by default.

- Main mode is slower, using more exchanges, but it protects the identities of the communicating peers.
- Aggressive mode is faster, but does not protect the identities of the peers.

To disable aggressive mode, enter the following command:

```console
crypto ikev1 am-disable
```
For example:
```console
hostname(config)# crypto ikev1 am-disable
```

If you have disabled aggressive mode, and want to revert to back to it, use the `no` form of the command. For example:
```console
hostname(config)# no crypto ikev1 am-disable
```

---

### Determining an ID Method for IKEv1 and IKEv2 ISAKMP Peers

During ISAKMP Phase I negotiations, either IKEv1 or IKEv2, the peers must identify themselves to each other. You can choose the identification method from the following options:
Chapter 26      Configuring IPsec and ISAKMP

Configuring ISAKMP

The ASA 1000V uses the Phase I ID to send to the peer. This is true for all VPN scenarios except LAN-to-LAN IKEv1 connections in main mode that authenticate with preshared keys. The default setting is auto.

To change the peer identification method, enter the following command:

crypt isakmp identity {address | hostname | key-id id-string | auto}

For example, the following command sets the peer identification method to hostname:

hostname(config)# crypto isakmp identity hostname

Enabling IPsec over NAT-T

NAT-T lets IPsec peers establish a connection through a NAT device. It does this by encapsulating IPsec traffic in UDP datagrams, using port 4500, which provides NAT devices with port information. NAT-T auto-detects any NAT devices and only encapsulates IPsec traffic when necessary. This feature is disabled by default.

Note

When IPsec over TCP is enabled, it takes precedence over all other connection methods.

When you enable NAT-T, the ASA 1000V automatically opens port 4500 on all IPsec-enabled interfaces.

Using NAT-T

To use NAT-T, you must perform the following tasks:

Step 1
Enter the following command to enable IPsec over NAT-T globally on the ASA 1000V:

crypto isakmp nat-traversal

tkealive

The range for the natkeepalive argument is 10 to 3600 seconds. The default is 20 seconds.

For example, enter the following command to enable NAT-T and set the keepalive value to one hour.

hostname(config)# crypto isakmp nat-traversal 3600

Step 2
Select the before-encryption option for the IPsec fragmentation policy by entering this command:

hostname(config)# crypto isakmp fragmentation before-encryption
Using the Tunnel-group-map default-group Command

This option lets traffic travel across NAT devices that do not support IP fragmentation. It does not impede the operation of NAT devices that do support IP fragmentation.

Waiting for Active Sessions to Terminate Before Rebooting

You can schedule an ASA 1000V reboot to occur only when all active sessions have terminated voluntarily. This feature is disabled by default.

To enable waiting for all active sessions to voluntarily terminate before the ASA 1000V reboots, enter the following command:

```
crypto isakmp reload-wait
```

For example:

```
hostname(config)# crypto isakmp reload-wait
```

Use the `reload` command to reboot the ASA 1000V. If you set the `reload-wait` command, you can use the `reload quick` command to override the `reload-wait` setting. The `reload` and `reload-wait` commands are available in privileged EXEC mode; neither includes the `isakmp` prefix.

Alerting Peers Before Disconnecting

LAN-to-LAN sessions can drop for several reasons, such as an ASA 1000V shutdown or reboot, session idle timeout, maximum connection time exceeded, or administrator cut-off.

The ASA 1000V can notify qualified peers in LAN-to-LAN configurations of sessions that are about to be disconnected. The peer receiving the alert decodes the reason and displays it in the event log or in a pop-up pane. This feature is disabled by default.

Security appliances with alerts enabled are qualified peers.

To enable disconnect notification to IPsec peers, enter the `crypto isakmp disconnect-notify` command.

For example:

```
hostname(config)# crypto isakmp disconnect-notify
```

Using the Tunnel-group-map default-group Command

This command specifies a default tunnel group to use when the configuration does not specify a tunnel group.

The syntax is `tunnel-group-map [rule-index] default-group tunnel-group-name` where `rule-index` is the priority for the rule, and `tunnel-group-name` must be for a tunnel group that already exists.

Configuring IPsec

This section provides background information about IPsec and describes the procedures required to configure the ASA 1000V when using IPsec to implement a VPN. It contains the following topics:
Understanding IPsec Tunnels

IPsec tunnels are sets of SAs that the ASA 1000V establishes between peers. The SAs specify the protocols and algorithms to apply to sensitive data and also specify the keying material that the peers use. IPsec SAs control the actual transmission of user traffic. SAs are unidirectional, but are generally established in pairs (inbound and outbound).

The peers negotiate the settings to use for each SA. Each SA consists of the following:

- IKEv1 transform sets or IKEv2 proposals
- Crypto maps
- Access lists
- Tunnel groups
- Prefragmentation policies

Understanding IKEv1 Transform Sets and IKEv2 Proposals

An IKEv1 transform set or an IKEv2 proposal is a combination of security protocols and algorithms that define how the ASA 1000V protects data. During IPsec SA negotiations, the peers must identify a transform set or proposal that is the same at both peers. The ASA 1000V then applies the matching transform set or proposal to create an SA that protects data flows in the access list for that crypto map.

With IKEv1 transform sets, you set one value for each parameter. For IKEv2 proposals, you can configure multiple encryption and authentication types and multiple integrity algorithms for a single proposal. The ASA 1000V orders the settings from the most secure to the least secure and negotiates with the peer using that order. This allows you to potentially send a single proposal to convey all the allowed combinations instead of the need to send each allowed combination individually as with IKEv1.

The ASA 1000V tears down the tunnel if you change the definition of the transform set or proposal used to create its SA. See “Clearing Security Associations” for further information.

Note

If you clear or delete the only element in a transform set or proposal, the ASA 1000V automatically removes the crypto map references to it.
Defining Crypto Maps

Crypto maps define the IPsec policy to be negotiated in the IPsec SA. They include the following:

- Access list to identify the packets that the IPsec connection permits and protects.
- Peer identification.
- Local address for the IPsec traffic. (See “Applying Crypto Maps to Interfaces” for more details.)
- Up to 11 IKEv1 transform sets or IKEv2 proposals, with which to attempt to match the peer security settings.

A crypto map set consists of one or more crypto maps that have the same map name. You create a crypto map set when you create its first crypto map. The following command syntax creates or adds to a crypto map:

```
crypto map map-name seq-num match address access-list-name
```

You can continue to enter this command to add crypto maps to the crypto map set. In the following example, `mymap` is the name of the crypto map set to which you might want to add crypto maps:

```
crypto map mymap 10 match address 101
```

The sequence number (seq-num) shown in the syntax above distinguishes one crypto map from another one with the same name. The sequence number assigned to a crypto map also determines its priority among the other crypto maps within a crypto map set. The lower the sequence number, the higher the priority. After you assign a crypto map set to an interface, the ASA 1000V evaluates all IP traffic passing through the interface against the crypto maps in the set, beginning with the crypto map with the lowest sequence number.

The ACL assigned to a crypto map consists of all of the ACEs that have the same access list name, as shown in the following command syntax:

```
access-list access-list-name (deny | permit) ip source source-netmask destination destination-netmask
```

Each ACL consists of one or more ACEs that have the same access list name. You create an ACL when you create its first ACE. The following command syntax creates or adds to an ACL:

```
access-list access-list-name (deny | permit) ip source source-netmask destination destination-netmask
```

In the following example, the ASA 1000V applies the IPsec protections assigned to the crypto map to all traffic flowing from the 10.0.0.0 subnet to the 10.1.1.0 subnet:

```
access-list 101 permit ip 10.0.0.0 255.255.255.0 10.1.1.0 255.255.255.0
```

The crypto map that matches the packet determines the security settings used in the SA negotiations. If the local ASA 1000V initiates the negotiation, it uses the policy specified in the static crypto map to create the offer to send to the specified peer. If the peer initiates the negotiation, the ASA 1000V attempts to match the policy to a static crypto map, and if that fails, then it attempts to match any dynamic crypto maps in the crypto map set, to decide whether to accept or reject the peer offer.

For two peers to succeed in establishing an SA, they must have at least one compatible crypto map. To be compatible, a crypto map must meet the following criteria:

- The crypto map must contain compatible crypto ACLs (for example, mirror image ACLs). If the responding peer uses dynamic crypto maps, so the ASA 1000V also must contain compatible crypto ACLs as a requirement to apply IPsec.
- Each crypto map identifies the other peer (unless the responding peer uses dynamic crypto maps).
The crypto maps have at least one transform set or proposal in common.

You can apply only one crypto map set to a single interface. Create more than one crypto map for a particular interface on the ASA 1000V if any of the following conditions exist:

- You want specific peers to handle different data flows.
- You want different IPsec security to apply to different types of traffic.

For example, create a crypto map and assign an ACL to identify traffic between two subnets and assign one IKEv1 transform set or IKEv2 proposal. Create another crypto map with a different ACL to identify traffic between another two subnets and apply a transform set or proposal with different VPN parameters.

If you create more than one crypto map for an interface, specify a sequence number (seq-num) for each map entry to determine its priority within the crypto map set.

Each ACE contains a permit or deny statement. Table 26-3 explains the special meanings of permit and deny ACEs in ACLs applied to crypto maps.

<table>
<thead>
<tr>
<th>Result of Crypto Map Evaluation</th>
<th>Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Match criterion in an ACE</td>
<td></td>
</tr>
<tr>
<td>containing a permit statement</td>
<td>Halt further evaluation of the packet against the remaining ACEs in the crypto map set, and evaluate the packet security settings against those in the IKEv1 transform sets or IKEv2 proposals assigned to the crypto map. After matching the security settings to those in a transform set or proposal, the ASA 1000V applies the associated IPsec settings. Typically for outbound traffic, this means that it decrypts, authenticates, and routes the packet.</td>
</tr>
<tr>
<td>Match criterion in an ACE</td>
<td></td>
</tr>
<tr>
<td>containing a deny statement</td>
<td>Interrupt further evaluation of the packet against the remaining ACEs in the crypto map under evaluation, and resume evaluation against the ACEs in the next crypto map, as determined by the next seq-num assigned to it.</td>
</tr>
<tr>
<td>Fail to match all tested permit</td>
<td>Route the packet without encrypting it.</td>
</tr>
<tr>
<td>ACEs in the crypto map set</td>
<td></td>
</tr>
</tbody>
</table>

ACEs containing deny statements filter out outbound traffic that does not require IPsec protection (for example, routing protocol traffic). Therefore, insert initial deny statements to filter outbound traffic that should not be evaluated against permit statements in a crypto access list.

For an inbound, encrypted packet, the security appliance uses the source address and ESP SPI to determine the decryption parameters. After the security appliance decrypts the packet, it compares the inner header of the decrypted packet to the permit ACEs in the ACL associated with the packet SA. If the inner header fails to match the proxy, the security appliance drops the packet. If the inner header matches the proxy, the security appliance routes the packet.

When comparing the inner header of an inbound packet that was not encrypted, the security appliance ignores all deny rules because they would prevent the establishment of a Phase 2 SA.

<table>
<thead>
<tr>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>To route inbound, unencrypted traffic as clear text, insert deny ACEs before permit ACEs.</td>
</tr>
</tbody>
</table>

Figure 26-1 shows an example LAN-to-LAN network of ASA 1000Vs.
The simple address notation shown in this figure and used in the following explanation is an abstraction. An example with real IP addresses follows the explanation.

The objective in configuring Security Appliances A, B, and C in this example LAN-to-LAN network is to permit tunneling of all traffic originating from one of the hosts shown in Figure 26-1 and destined for one of the other hosts. However, because traffic from Host A.3 contains sensitive data from the Human Resources department, it requires strong encryption and more frequent rekeying than the other traffic. So you will want to assign a special transform set for traffic from Host A.3.

To configure Security Appliance A for outbound traffic, you create two crypto maps, one for traffic from Host A.3 and the other for traffic from the other hosts in Network A, as shown in the following example:

```
Crypto Map Seq_No_1
  deny packets from A.3 to B
  deny packets from A.3 to C
  permit packets from A to B
  permit packets from A to C
```

```
Crypto Map Seq_No_2
  permit packets from A.3 to B
  permit packets from A.3 to C
```

After creating the ACLs, you assign a transform set to each crypto map to apply the required IPsec to each matching packet.

Cascading ACLs involves the insertion of deny ACEs to bypass evaluation against an ACL and resume evaluation against a subsequent ACL in the crypto map set. Because you can associate each crypto map with different IPsec settings, you can use deny ACEs to exclude special traffic from further evaluation in the corresponding crypto map, and match the special traffic to permit statements in another crypto map to provide or require different security. The sequence number assigned to the crypto ACL determines its position in the evaluation sequence within the crypto map set.
Figure 26-2 shows the cascading ACLs created from the conceptual ACEs above. The meaning of each symbol in the figure follows.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Symbol" /></td>
<td>Crypto map within a crypto map set.</td>
</tr>
<tr>
<td><img src="image2" alt="Symbol" /></td>
<td>(Gap in a straight line) Exit from a crypto map when a packet matches an ACE.</td>
</tr>
<tr>
<td><img src="image3" alt="Symbol" /></td>
<td>Packet that fits the description of one ACE. Each size ball represents a different packet matching the respective ACE in the figure. The differences in size merely represent differences in the source and destination of each packet.</td>
</tr>
<tr>
<td><img src="image4" alt="Symbol" /></td>
<td>Redirection to the next crypto map in the crypto map set.</td>
</tr>
<tr>
<td><img src="image5" alt="Symbol" /></td>
<td>Response when a packet either matches an ACE or fails to match all of the permit ACEs in a crypto map set.</td>
</tr>
</tbody>
</table>
Security Appliance A evaluates a packet originating from Host A.3 until it matches a permit ACE and attempts to assign the IPsec security associated with the crypto map. Whenever the packet matches a deny ACE, the ASA 1000V ignores the remaining ACEs in the crypto map and resumes evaluation against the next crypto map, as determined by the sequence number assigned to it. So in the example, if Security Appliance A receives a packet from Host A.3, it matches the packet to a deny ACE in the first crypto map and resumes evaluation of the packet against the next crypto map. When it matches the packet to the permit ACE in that crypto map, it applies the associated IPsec security (strong encryption and frequent rekeying).
To complete the security appliance configuration in the example network, we assign mirror crypto maps to Security Appliances B and C. However, because security appliances ignore deny ACEs when evaluating inbound, encrypted traffic, we can omit the mirror equivalents of the deny A.3 B and deny A.3 C ACEs, and therefore omit the mirror equivalents of Crypto Map 2. So the configuration of cascading ACLs in Security Appliances B and C is unnecessary.

Table 26-4 shows the ACLs assigned to the crypto maps configured for all three ASA 1000Vs in Figure 26-1.

<table>
<thead>
<tr>
<th>Security Appliance A</th>
<th>Security Appliance B</th>
<th>Security Appliance C</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Crypto Map Sequence No.</strong></td>
<td><strong>ACE Pattern</strong></td>
<td><strong>Crypto Map Sequence No.</strong></td>
</tr>
<tr>
<td>1</td>
<td>deny A.3 B</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>deny A.3 C</td>
<td></td>
</tr>
<tr>
<td></td>
<td>permit A B</td>
<td></td>
</tr>
<tr>
<td></td>
<td>permit A C</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>permit A.3 B</td>
<td></td>
</tr>
<tr>
<td></td>
<td>permit A.3 C</td>
<td></td>
</tr>
</tbody>
</table>

Figure 26-3 maps the conceptual addresses shown in Figure 26-1 to real IP addresses.
The tables that follow combine the IP addresses shown in Figure 26-3 to the concepts shown in Table 26-4. The real ACEs shown in these tables ensure that all IPsec packets under evaluation within this network receive the proper IPsec settings.

### Table 26-5 Example Permit and Deny Statements for Security Appliance A

<table>
<thead>
<tr>
<th>Security Appliance</th>
<th>Crypto Map Sequence No.</th>
<th>ACE Pattern</th>
<th>Real ACEs</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1</td>
<td>deny A.3 B</td>
<td>deny 192.168.3.3 255.255.255.192 192.168.12.0 255.255.255.248</td>
</tr>
<tr>
<td></td>
<td></td>
<td>deny A.3 C</td>
<td>deny 192.168.3.3 255.255.255.192 192.168.201.0 255.255.255.224</td>
</tr>
<tr>
<td></td>
<td></td>
<td>permit A B</td>
<td>permit 192.168.3.0 255.255.255.192 192.168.12.0 255.255.255.248</td>
</tr>
<tr>
<td></td>
<td></td>
<td>permit A C</td>
<td>permit 192.168.3.0 255.255.255.192 192.168.201.0 255.255.255.224</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>permit A.3 B</td>
<td>permit 192.168.3.3 255.255.255.192 192.168.12.0 255.255.255.248</td>
</tr>
<tr>
<td></td>
<td></td>
<td>permit A.3 C</td>
<td>permit 192.168.3.3 255.255.255.192 192.168.201.0 255.255.255.224</td>
</tr>
<tr>
<td>B</td>
<td>None needed</td>
<td>permit B A</td>
<td>permit 192.168.12.0 255.255.255.248 192.168.3.0 255.255.255.192</td>
</tr>
<tr>
<td></td>
<td></td>
<td>permit B C</td>
<td>permit 192.168.12.0 255.255.255.248 192.168.201.0 255.255.255.224</td>
</tr>
<tr>
<td>C</td>
<td>None needed</td>
<td>permit C A</td>
<td>permit 192.168.201.0 255.255.255.224 192.168.3.0 255.255.255.192</td>
</tr>
<tr>
<td></td>
<td></td>
<td>permit C B</td>
<td>permit 192.168.201.0 255.255.255.224 192.168.12.0 255.255.255.248</td>
</tr>
</tbody>
</table>

You can apply the same reasoning shown in the example network to use cascading ACLs to assign different security settings to different hosts or subnets protected by a Cisco ASA 1000V.
By default, the ASA 1000V does not support IPsec traffic destined for the same interface from which it enters. Names for this type of traffic include U-turn, hub-and-spoke, and hairpinning. However, you can configure IPsec to support U-turn traffic by inserting an ACE to permit traffic to and from the network. For example, to support U-turn traffic on Security Appliance B, add a conceptual “permit B B” ACE to ACL1. The actual ACE would be as follows:

```
permit 192.168.12.0 255.255.255.248 192.168.12.0 255.255.255.248
```

### Applying Crypto Maps to Interfaces

You must assign a crypto map set to each interface through which IPsec traffic flows. The ASA 1000V supports IPsec on all interfaces. Assigning the crypto map set to an interface instructs the ASA 1000V to evaluate all the traffic against the crypto map set and to use the specified policy during connection or SA negotiation.

Assigning a crypto map to an interface also initializes run-time data structures, such as the SA database and the security policy database. Reassigning a modified crypto map to the interface resynchronizes the run-time data structures with the crypto map configuration. Also, adding new peers through the use of new sequence numbers and reassigning the crypto map does not tear down existing connections.

### Using Interface Access Lists

By default, the ASA 1000V lets IPsec packets bypass interface ACLs. If you want to apply interface access lists to IPsec traffic, use the `no` form of the `sysopt connection permit-vpn` command.

The crypto map access list bound to the outgoing interface either permits or denies IPsec packets through the VPN tunnel. IPsec authenticates and deciphers packets that arrive from an IPsec tunnel, and subjects them to evaluation against the ACL associated with the tunnel.

Access lists define which IP traffic to protect. For example, you can create access lists to protect all IP traffic between two subnets or two hosts. (These access lists are similar to access lists used with the `access-group` command. However, with the `access-group` command, the access list determines which traffic to forward or block at an interface.)

Before the assignment to crypto maps, the access lists are not specific to IPsec. Each crypto map references the access lists and determines the IPsec properties to apply to a packet if it matches a permit in one of the access lists.

Access lists assigned to IPsec crypto maps have four primary functions:

- Select outbound traffic to be protected by IPsec (permit = protect).
- Trigger an ISAKMP negotiation for data travelling without an established SA.
- Process inbound traffic to filter out and discard traffic that should have been protected by IPsec.
- Determine whether to accept requests for IPsec SAs when processing IKE negotiation from the peer. (Negotiation applies only to `ipsec-isakmp crypto map` entries.) The peer must permit a data flow associated with an `ipsec-isakmp crypto map` command entry to ensure acceptance during negotiation.

Regardless of whether the traffic is inbound or outbound, the ASA 1000V evaluates traffic against the access lists assigned to an interface. You assign IPsec to an interface as follows:

**Step 1** Create the access lists to be used for IPsec.
**Step 2** Map the lists to one or more crypto maps, using the same crypto map name.

**Step 3** Map the IKEv1 transform sets or IKEv2 proposals to the crypto maps to apply IPsec to the data flows.

**Step 4** Apply the crypto maps collectively as a crypto map set by assigning the crypto map name they share to the interface.

In Figure 26-4, IPsec protection applies to traffic between Host 10.0.0.1 and Host 10.2.2.2 as the data exits the outside interface on Security Appliance A toward Host 10.2.2.2.

**Figure 26-4 How Crypto Access Lists Apply to IPsec**

Security Appliance A evaluates traffic from Host 10.0.0.1 to Host 10.2.2.2, as follows:

- source = host 10.0.0.1
- dest = host 10.2.2.2

Security Appliance A also evaluates traffic from Host 10.2.2.2 to Host 10.0.0.1, as follows:

- source = host 10.2.2.2
- dest = host 10.0.0.1

The first permit statement that matches the packet under evaluation determines the scope of the IPsec SA.

**Note**

If you delete the only element in an access list, the ASA 1000V also removes the associated crypto map.

If you modify an access list currently referenced by one or more crypto maps, use the **crypto map interface** command to reinitialize the run-time SA database. See the **crypto map** command for more information.

We recommend that for every crypto access list specified for a static crypto map that you define at the local peer, you define a “mirror image” crypto access list at the remote peer. The crypto maps should also support common transforms and refer to the other system as a peer. This ensures correct processing of IPsec by both peers.
Note
Every static crypto map must define an access list and an IPsec peer. If either is missing, the crypto map is incomplete and the ASA 1000V drops any traffic that it has not already matched to an earlier, complete crypto map. Use the `show conf` command to ensure that every crypto map is complete. To fix an incomplete crypto map, remove the crypto map, add the missing entries, and reapply it.

We discourage the use of the `any` keyword to specify source or destination addresses in crypto access lists because they cause problems. We strongly discourage the `permit any any` command because it does the following:

- Protects all outbound traffic, including all protected traffic sent to the peer specified in the corresponding crypto map.
- Requires protection for all inbound traffic.

In this scenario, the ASA 1000V silently drops all inbound packets that lack IPsec protection. Be sure that you define which packets to protect. If you use the `any` keyword in a `permit` statement, preface it with a series of `deny` statements to filter out traffic that would otherwise fall within that `permit` statement that you do not want to protect.

Note
Decrypted through traffic is permitted from the client despite having an access group on the outside interface, which calls a deny ip any any access-list, while `no sysopt connection permit-vpn` is configured.

Users who want to control access to the protected network via site-to-site or remote access VPN using the `no sysopt permit` command in conjunction with an access control list (ACL) on the outside interface are not successful.

In this situation, when management-access inside is enabled, the ACL is not applied, and users can still connect using SSH to the security appliance. Traffic to hosts on the inside network are blocked correctly by the ACL, but cannot block decrypted through traffic to the inside interface.

The `ssh` and `http` commands are of a higher priority than the ACLs. In other words, to deny SSH, Telnet, or ICMP traffic to the device from the VPN session, use `ssh`, `telnet` and `icmp` commands, which deny the IP local pool should be added.

### Changing IPsec SA Lifetimes

You can change the global lifetime values that the ASA 1000V uses when negotiating new IPsec SAs. You can override these global lifetime values for a particular crypto map.

IPsec SAs use a derived, shared, secret key. The key is an integral part of the SA; the keys time out together to require the key to refresh. Each SA has two lifetimes: timed and traffic-volume. An SA expires after the respective lifetime and negotiations begin for a new one. The default lifetimes are 28,800 seconds (eight hours) and 4,608,000 kilobytes (10 megabytes per second for one hour).

If you change a global lifetime, the ASA 1000V drops the tunnel. It uses the new value in the negotiation of subsequently established SAs.

When a crypto map does not have configured lifetime values and the ASA 1000V requests a new SA, it inserts the global lifetime values used in the existing SA into the request sent to the peer. When a peer receives a negotiation request, it uses the smaller of either the lifetime value the peer proposes or the locally configured lifetime value as the lifetime of the new SA.
The peers negotiate a new SA before crossing the lifetime threshold of the existing SA to ensure that a new SA is ready when the existing one expires. The peers negotiate a new SA when about 5 to 15 percent of the lifetime of the existing SA remains.

Creating a Basic IPsec Configuration

You can create basic IPsec configurations with static or dynamic crypto maps.

To create a basic IPsec configuration using a static crypto map, perform the following steps:

**Step 1**
To create an access list to define the traffic to protect, enter the following command:
```
access-list access-list-name (deny | permit) ip source source-netmask destination destination-netmask
```
For example:
```
hostname(config)# access-list 101 permit ip 10.0.0.0 255.255.255.0 10.1.1.0 255.255.255.0
```
In this example, the permit keyword causes all traffic that matches the specified conditions to be protected by crypto.

**Step 2**
To configure an IKEv1 transform set that defines how to protect the traffic, enter the following command:
```
crypto ipsec ikev1 transform-set transform-set-name encryption [authentication]
```
For example:
```
hostname(config)# crypto ipsec ikev1 transform-set myset1 esp-des esp-sha-hmac
hostname(config)# crypto ipsec ikev1 transform-set myset2 esp-3des esp-sha-hmac
hostname(config)# crypto ipsec ikev1 transform-set aes_set esp-md5-hmac esp-aes-256
```
In this example, myset1 and myset2 and aes_set are the names of the transform sets.

To configure an IKEv2 proposal that also defines how to protect the traffic, enter the `crypto ipsec ikev2 ipsec-proposal` command to create the proposal and enter the ipsec proposal configuration mode where you can specify multiple encryption and integrity types for the proposal:
```
crypto ipsec ikev2 ipsec-proposal [proposal tag]
```
For example:
```
hostname(config)# crypto ipsec ikev2 ipsec-proposal secure
```
In this example, secure is the name of the proposal. Enter a protocol and encryption types:
```
hostname(config-ipsec-proposal)# protocol esp encryption 3des aes des
```

**Step 3**
To create a crypto map, perform the following steps:

a. Assign an access list to a crypto map:
```
crypto map map-name seq-num match address access-list-name
```
In the following example, mymap is the name of the crypto map set. The map set sequence number 10, which is used to rank multiple entries within one crypto map set. The lower the sequence number, the higher the priority.
```
crypto map mymap 10 match address 101
```
In this example, the access list named 101 is assigned to crypto map mymap.
b. Specify the peer to which the IPsec-protected traffic can be forwarded:

```
crypto map map-name seq-num set peer ip-address
```

For example:
```
crypto map mymap 10 set peer 192.168.1.100
```

The ASA 1000V sets up an SA with the peer assigned the IP address 192.168.1.100.
Specify multiple peers by repeating this command.

c. Specify which IKEv1 transform sets or IKEv2 proposals are allowed for this crypto map. List multiple transform sets or proposals in order of priority (highest priority first). You can specify up to 11 transform sets or proposals in a crypto map using either of these two commands:

```
crypto map map-name seq-num set ikev1 transform-set transform-set-name1
[transform-set-name2, ...transform-set-name11]
crypto map map-name seq-num set ikev2 ipsec-proposal proposal-name1
[proposal-name2, ... proposal-name11]
```

For example (for IKEv1):
```
crypto map mymap 10 set ikev1 transform-set myset1 myset2
```

In this example, when traffic matches access list 101, the SA can use either myset1 (first priority) or myset2 (second priority) depending on which transform set matches the transform set of the peer.

d. (Optional) Specify an SA lifetime for the crypto map if you want to override the global lifetime.

```
crypto map map-name seq-num set security-association lifetime {seconds seconds | kilobytes kilobytes}
```

For example:
```
crypto map mymap 10 set security-association lifetime seconds 2700
```

This example shortens the timed lifetime for the crypto map mymap 10 to 2700 seconds (45 minutes). The traffic volume lifetime is not changed.

e. (Optional) Specify that IPsec require perfect forward secrecy when requesting new SA for this crypto map, or require PFS in requests received from the peer:

```
crypto map map-name seq-num set pfs [group1 | group2 | group5]
```

For example:
```
crypto map mymap 10 set pfs group2
```

This example requires PFS when negotiating a new SA for the crypto map mymap 10. The ASA 1000V uses the 1024-bit Diffie-Hellman prime modulus group in the new SA.

**Step 4** Apply a crypto map set to an interface for evaluating IPsec traffic:

```
crypto map map-name interface interface-name
```

For example:
```
crypto map mymap interface outside
```

In this example, the ASA 1000V evaluates the traffic going through the outside interface against the crypto map mymap to determine whether it needs to be protected.
Using Dynamic Crypto Maps

A dynamic crypto map is a crypto map without all of the parameters configured. It acts as a policy template where the missing parameters are later dynamically learned, as the result of an IPsec negotiation, to match the peer requirements. The ASA 1000V applies a dynamic crypto map to let a peer negotiate a tunnel if its IP address is not already identified in a static crypto map. This occurs with the following types of peers:

- Peers with dynamically assigned public IP addresses.
  LAN-to-LAN peers can use DHCP to obtain a public IP address. The ASA 1000V uses this address only to initiate the tunnel.
- Peers with dynamically assigned private IP addresses.
  Peers requesting remote access tunnels typically have private IP addresses assigned by the headend. Generally, LAN-to-LAN tunnels have a predetermined set of private networks that are used to configure static maps and therefore used to establish IPsec SAs.

As an administrator configuring static crypto maps, you might not know the IP addresses that are dynamically assigned (via DHCP or some other method), and you might not know the private IP addresses of other clients, regardless of how they were assigned.

---

**Note**

A dynamic crypto map requires only the `transform-set` parameter.

Dynamic crypto maps can ease IPsec configuration, and we recommend them for use in networks where the peers are not always predetermined. Use dynamic crypto maps for routers that obtain dynamically assigned IP addresses.

---

**Tip**

Use care when using the `any` keyword in `permit` entries in dynamic crypto maps. If the traffic covered by such a `permit` entry could include multicast or broadcast traffic, insert `deny` entries for the appropriate address range into the access list. Remember to insert `deny` entries for network and subnet broadcast traffic, and for any other traffic that IPsec should not protect.

Dynamic crypto maps work only to negotiate SAs with remote peers that initiate the connection. The ASA 1000V cannot use dynamic crypto maps to initiate connections to a remote peer. With a dynamic crypto map, if outbound traffic matches a permit entry in an access list and the corresponding SA does not yet exist, the ASA 1000V drops the traffic.

A crypto map set may include a dynamic crypto map. Dynamic crypto map sets should be the lowest priority crypto maps in the crypto map set (that is, they should have the highest sequence numbers) so that the ASA 1000V evaluates other crypto maps first. It examines the dynamic crypto map set only when the other (static) map entries do not match.

Similar to static crypto map sets, a dynamic crypto map set consists of all of the dynamic crypto maps with the same dynamic-map-name. The dynamic-seq-num differentiates the dynamic crypto maps in a set. If you configure a dynamic crypto map, insert a permit ACL to identify the data flow of the IPsec peer for the crypto access list. Otherwise the ASA 1000V accepts any data flow identity the peer proposes.
Caution

Do not assign module default routes for traffic to be tunneled to a ASA 1000V interface configured with a dynamic crypto map set. To identify the traffic that should be tunneled, add the ACLs to the dynamic crypto map. Use care to identify the proper address pools when configuring the ACLs associated with remote access tunnels. Use Reverse Route Injection to install routes only after the tunnel is up.

The procedure for using a dynamic crypto map entry is the same as the basic configuration described in “Creating a Basic IPsec Configuration,” except that instead of creating a static crypto map, you create a dynamic crypto map entry. You can also combine static and dynamic map entries within a single crypto map set.

Create a crypto dynamic map entry as follows:

Step 1  (Optional) Assign an access list to a dynamic crypto map:

```
crypto dynamic-map dynamic-map-name dynamic-seq-num match address access-list-name
```

This determines which traffic should be protected and not protected.

For example:

```
crypto dynamic-map dyn1 10 match address 101
```

In this example, access list 101 is assigned to dynamic crypto map dyn1. The map sequence number is 10.

Step 2  Specify which IKEv1 transform sets or IKEv2 proposals are allowed for this dynamic crypto map. List multiple transform sets or proposals in order of priority (highest priority first) using the command for IKEv1 transform sets or IKEv2 proposals:

```
crypto dynamic-map dynamic-map-name dynamic-seq-num set ikev1 transform-set
transform-set-name1, [transform-set-name2, ...transform-set-name9]
crypto dynamic-map dynamic-map-name dynamic-seq-num set ikev2 ipsec-proposal
proposal-name1 [proposal-name2, ... proposal-name11]
```

For example (for IKEv1):

```
crypto dynamic-map dyn 10 set ikev1 transform-set myset1 myset2
```

In this example, when traffic matches access list 101, the SA can use either myset1 (first priority) or myset2 (second priority), depending on which transform set matches the transform sets of the peer.

Step 3  (Optional) Specify the SA lifetime for the crypto dynamic map entry if you want to override the global lifetime value:

```
crypto dynamic-map dynamic-map-name dynamic-seq-num set security-association lifetime
{seconds seconds | kilobytes kilobytes}
```

For example:

```
crypto dynamic-map dyn1 10 set security-association lifetime seconds 2700
```

This example shortens the timed lifetime for dynamic crypto map dyn1 10 to 2700 seconds (45 minutes). The time volume lifetime is not changed.

Step 4  (Optional) Specify that IPsec ask for PFS when requesting new SAs for this dynamic crypto map, or should demand PFS in requests received from the peer:

```
crypto dynamic-map dynamic-map-name dynamic-seq-num set pfs [group1 | group2 | group5 | group7]
```
For example:

```plaintext
crypto dynamic-map dyn1 10 set pfs group5
```

**Step 5**

Add the dynamic crypto map set into a static crypto map set.

Be sure to set the crypto maps referencing dynamic maps to be the lowest priority entries (highest sequence numbers) in a crypto map set.

```plaintext
crypto map map-name seq-num ipsec-isakmp dynamic dynamic-map-name
```

For example:

```plaintext
crypto map mymap 200 ipsec-isakmp dynamic dyn1
```

---

**Providing Site-to-Site Redundancy**

You can define multiple IKEv1 peers by using crypto maps to provide redundancy. This configuration is useful for site-to-site VPNs. This feature is not supported with IKEv2.

If one peer fails, the ASA 1000V establishes a tunnel to the next peer associated with the crypto map. It sends data to the peer that it has successfully negotiated with, and that peer becomes the active peer. The active peer is the peer that the ASA 1000V keeps trying first for follow-on negotiations until a negotiation fails. At that point the ASA 1000V goes on to the next peer. The ASA 1000V cycles back to the first peer when all peers associated with the crypto map have failed.

---

**Viewing an IPsec Configuration**

Table 26-6 lists commands that you can enter to view information about your IPsec configuration.

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show running-configuration crypto</td>
<td>Displays the entire crypto configuration, including IPsec, crypto maps, dynamic crypto maps, and ISAKMP.</td>
</tr>
<tr>
<td>show running-config crypto ipsec</td>
<td>Displays the complete IPsec configuration.</td>
</tr>
<tr>
<td>show running-config crypto isakmp</td>
<td>Displays the complete ISAKMP configuration.</td>
</tr>
<tr>
<td>show running-config crypto map</td>
<td>Displays the complete crypto map configuration.</td>
</tr>
<tr>
<td>show running-config crypto dynamic-map</td>
<td>Displays the dynamic crypto map configuration.</td>
</tr>
<tr>
<td>show all crypto map</td>
<td>Displays all of the configuration parameters, including those with default values.</td>
</tr>
</tbody>
</table>
Clearing Security Associations

Certain configuration changes take effect only during the negotiation of subsequent SAs. If you want the new settings to take effect immediately, clear the existing SAs to reestablish them with the changed configuration. If the ASA 1000V is actively processing IPsec traffic, clear only the portion of the SA database that the configuration changes affect. Reserve clearing the full SA database for large-scale changes, or when the ASA 1000V is processing a small amount of IPsec traffic.

Table 26-7 lists commands you can enter to clear and reinitialize IPsec SAs.

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>clear configure crypto</td>
<td>Removes an entire crypto configuration, including IPsec, crypto maps, dynamic crypto maps, and ISAKMP.</td>
</tr>
<tr>
<td>clear configure crypto ca trustpoint</td>
<td>Removes all trustpoints.</td>
</tr>
<tr>
<td>clear configure crypto dynamic-map</td>
<td>Removes all dynamic crypto maps. Includes keywords that let you remove specific dynamic crypto maps.</td>
</tr>
<tr>
<td>clear configure crypto map</td>
<td>Removes all crypto maps. Includes keywords that let you remove specific crypto maps.</td>
</tr>
<tr>
<td>clear configure crypto isakmp</td>
<td>Removes the entire ISAKMP configuration.</td>
</tr>
<tr>
<td>clear configure crypto isakmp policy</td>
<td>Removes all ISAKMP policies or a specific policy.</td>
</tr>
<tr>
<td>clear crypto isakmp sa</td>
<td>Removes the entire ISAKMP SA database.</td>
</tr>
</tbody>
</table>

Clearing Crypto Map Configurations

The clear configure crypto command includes arguments that let you remove elements of the crypto configuration, including IPsec, crypto maps, dynamic crypto maps, CA trustpoints, all certificates, certificate map configurations, and ISAKMP.

Be aware that if you enter the clear configure crypto command without arguments, you remove the entire crypto configuration, including all certificates.

For more information, see the clear configure crypto command in the command reference.
CHAPTER 27

Configuring Connection Profiles, Group Policies, and Users

This chapter describes how to configure VPN connection profiles (formerly called “tunnel groups”), group policies, and users. This chapter includes the following sections.

- Overview of Connection Profiles, Group Policies, and Users, page 27-1
- Configuring Connection Profiles, page 27-3
- Group Policies, page 27-7

In summary, you first configure connection profiles to set the values for the connection. Then you configure group policies. These set values for users in the aggregate. Then you configure users, which can inherit values from groups and configure certain values on an individual user basis. This chapter describes how and why to configure these entities.

Overview of Connection Profiles, Group Policies, and Users

Groups and users are core concepts in managing the security of virtual private networks (VPNs) and in configuring the ASA 1000V. They specify attributes that determine user access to and use of the VPN. A group is a collection of users treated as a single entity. Users get their attributes from group policies. A connection profile identifies the group policy for a specific connection. If you do not assign a particular group policy to a user, the default group policy for the connection applies.

You configure connection profiles using tunnel-group commands. In this chapter, the terms “connection profile” and “tunnel group” are often used interchangeably.

Connection profiles and group policies simplify system management. To streamline the configuration task, the ASA 1000V provides a default LAN-to-LAN connection profile, a default remote access connection profile, a default connection profile for SSL/IKEv2 VPN, and a default group policy (DfltGrpPolicy). The default connection profiles and group policy provide settings that are likely to be common for many users. As you add users, you can specify that they “inherit” parameters from a group policy. Thus you can quickly configure VPN access for large numbers of users.

If you decide to grant identical rights to all VPN users, then you do not need to configure specific connection profiles or group policies, but VPNs seldom work that way. For example, you might allow a finance group to access one part of a private network, a customer support group to access another part,
Connection Profiles

A connection profile consists of a set of records that determines tunnel connection policies. These records identify the servers to which the tunnel user is authenticated, as well as the accounting servers, if any, to which connection information is sent. They also identify a default group policy for the connection, and they contain protocol-specific connection parameters. Connection profiles include a small number of attributes that pertain to creating the tunnel itself. Connection profiles include a pointer to a group policy that defines user-oriented attributes.

The ASA 1000V provides a default DefaultL2Lgroup for LAN-to-LAN connection profiles. You can modify these default connection profiles, but you cannot delete them. You can also create one or more connection profiles specific to your environment. Connection profiles are local to the ASA 1000V and are not configurable on external servers.

General Connection Profile Connection Parameters

General parameters are common to all VPN connections. The general parameters include the following:

- Connection profile name—You specify a connection-profile name when you add or edit a connection profile.
- Connection type—Connection types include IPsec Lan-to-LAN.
• Authentication, Authorization, and Accounting servers—These parameters identify the server groups or lists that the ASA 1000V uses for the following purposes:
  – Authenticating users
  – Obtaining information about services users are authorized to access
  – Storing accounting records
A server group can consist of one or more servers.
• Default group policy for the connection—A group policy is a set of user-oriented attributes. The default group policy is the group policy whose attributes the ASA 1000V uses as defaults when authenticating or authorizing a tunnel user.

Configuring Connection Profiles

The following sections describe the contents and configuration of connection profiles:
• Maximum Connection Profiles, page 27-3
• Configuring LAN-to-LAN Connection Profiles, page 27-4
You can modify the default connection profiles, and you can configure a new connection profile as any of the three tunnel-group types. If you don’t explicitly configure an attribute in a connection profile, that attribute gets its value from the default connection profile. The default connection-profile type is remote access. The subsequent parameters depend upon your choice of tunnel type. To see the current configured and default configuration of all your connection profiles, including the default connection profile, enter the `show running-config all tunnel-group` command.

Maximum Connection Profiles

The maximum number of connection profiles (tunnel groups) that an ASA 1000V can support is a function of the maximum number of concurrent VPN sessions for the platform + 5. For example, an ASA5505 can support a maximum of 25 concurrent VPN sessions allowing for 30 tunnel groups (25+5). Attempting to add an additional tunnel group beyond the limit results in the following message: “ERROR: The limit of 30 configured tunnel groups has been reached”.

Table Table 27-1 specifies the maximum VPN sessions and connection profiles for each ASA platform.

<table>
<thead>
<tr>
<th>Table 27-1</th>
<th>Maximum VPN Sessions and Connection Profiles Per ASA Platform</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum VPN Sessions</td>
<td>10/25</td>
</tr>
<tr>
<td>Maximum Connection Profiles</td>
<td>15/30</td>
</tr>
</tbody>
</table>

Configuring IPsec Tunnel-Group General Attributes

IPsec LAN-to-LAN tunnels use a subset of tunnel-group attributes. Refer to the command reference for complete descriptions of all commands. The following sections describe, in order, how to configure LAN-to-LAN connection profiles.
Configuring LAN-to-LAN Connection Profiles

An IPsec LAN-to-LAN VPN connection profile applies only to LAN-to-LAN IPsec client connections. While many of the parameters that you configure are the same as for IPsec remote-access connection profiles, LAN-to-LAN tunnels have fewer parameters. The following sections show you how to configure a LAN-to-LAN connection profile:

- Specifying a Name and Type for a LAN-to-LAN Connection Profile, page 27-4
- Configuring LAN-to-LAN Connection Profile General Attributes, page 27-4
- Configuring LAN-to-LAN IPsec IKEv1 Attributes, page 27-5

Default LAN-to-LAN Connection Profile Configuration

The contents of the default LAN-to-LAN connection profile are as follows:

tunnel-group DefaultL2LGroup type ipsec-l2l
tunnel-group DefaultL2LGroup general-attributes
  no accounting-server-group
  default-group-policy DfltGrpPolicy
tunnel-group DefaultL2LGroup ipsec-attributes
  no ikev1 pre-shared-key
  peer-id-validate req
  no chain
  no ikev1 trust-point
  isakmp keepalive threshold 10 retry 2

LAN-to-LAN connection profiles have fewer parameters than remote-access connection profiles, and most of these are the same for both groups. For your convenience in configuring the connection, they are listed separately here. Any parameters that you do not explicitly configure inherit their values from the default connection profile.

Specifying a Name and Type for a LAN-to-LAN Connection Profile

To specify a name and a type for a connection profile, enter the `tunnel-group` command, as follows:

```
hostname(config)# tunnel-group tunnel_group_name type tunnel_type
```

For a LAN-to-LAN tunnel, the type is `ipsec-l2l`; for example, to create the LAN-to-LAN connection profile named docs, enter the following command:

```
hostname(config)# tunnel-group docs type ipsec-l2l
```

Configuring LAN-to-LAN Connection Profile General Attributes

To configure the connection profile general attributes, do the following steps:

**Step 1**

Enter tunnel-group general-attributes mode by specifying the general-attributes keyword:

```
hostname(config)# tunnel-group_tunnel-group-name general-attributes
```

The prompt changes to indicate that you are now in config-general mode, in which you configure the tunnel-group general attributes.
For example, for the connection profile named docs, enter the following command:

```
hostname(config)# tunnel-group_docs general-attributes
hostname(config-tunnel-general)#
```

**Step 2** Specify the name of the accounting-server group, if any, to use:

```
hostname(config-tunnel-general)# accounting-server-group groupname
hostname(config-tunnel-general)#
```

For example, the following command specifies the use of the accounting-server group acctgserv1:

```
hostname(config-tunnel-general)# accounting-server-group acctgserv1
hostname(config-tunnel-general)#
```

**Step 3** Specify the name of the default group policy:

```
hostname(config-tunnel-general)# default-group-policy policyname
hostname(config-tunnel-general)#
```

For example, the following command specifies that the name of the default group policy is MyPolicy:

```
hostname(config-tunnel-general)# default-group-policy MyPolicy
hostname(config-tunnel-general)#
```

### Configuring LAN-to-LAN IPsec IKEv1 Attributes

To configure the IPsec IKEv1 attributes, do the following steps:

**Step 1** To configure the tunnel-group IPsec IKEv1 attributes, enter tunnel-group ipsec-attributes configuration mode by entering the tunnel-group command with the IPsec-attributes keyword.

```
hostname(config)# tunnel-group tunnel-group-name ipsec-attributes
hostname(config-tunnel-ipsec)#
```

For example, the following command enters config-ipsec mode so you can configure the parameters for the connection profile named TG1:

```
hostname(config)# tunnel-group TG1 ipsec-attributes
hostname(config-tunnel-ipsec)#
```

The prompt changes to indicate that you are now in tunnel-group ipsec-attributes configuration mode.

**Step 2** Specify the preshared key to support IKEv1 connections based on preshared keys.

```
hostname(config-tunnel-ipsec)# ikev1 pre-shared-key key
hostname(config-tunnel-ipsec)#
```

For example, the following command specifies the preshared key XYZX to support IKEv1 connections for an LAN-to-LAN connection profile:

```
hostname(config-tunnel-ipsec)# ikev1 pre-shared-key XYZX
hostname(config-tunnel-general)#
```

**Step 3** Specify whether to validate the identity of the peer using the peer’s certificate:

```
hostname(config-tunnel-ipsec)# peer-id-validate option
hostname(config-tunnel-ipsec)#
```
The available options are `req` (required), `cert` (if supported by certificate), and `nocheck` (do not check). The default is `req`. For example, the following command sets the peer-id-validate option to `nocheck`:

```
hostname(config-tunnel-ipsec)# peer-id-validate nocheck
hostname(config-tunnel-ipsec)#
```

**Step 4** Specify whether to enable sending of a certificate chain. This action includes the root certificate and any subordinate CA certificates in the transmission:

```
hostname(config-tunnel-ipsec)# chain
hostname(config-tunnel-ipsec)#
```

You can apply this attribute to all tunnel-group types.

**Step 5** Specify the name of a trustpoint that identifies the certificate to be sent to the IKE peer:

```
hostname(config-tunnel-ipsec)# trust-point trust-point-name
hostname(config-tunnel-ipsec)#
```

For example, the following command sets the trustpoint name to `mytrustpoint`:

```
hostname(config-tunnel-ipsec)# trust-point mytrustpoint
hostname(config-tunnel-ipsec)#
```

You can apply this attribute to all tunnel-group types.

**Step 6** Specify the ISAKMP (IKE) keepalive threshold and the number of retries allowed. The `threshold` parameter specifies the number of seconds (10 through 3600) that the peer is allowed to idle before beginning keepalive monitoring. The `retry` parameter is the interval (2 through 10 seconds) between retries after a keepalive response has not been received. IKE keepalives are enabled by default. To disable IKE keepalives, enter the `no` form of the `isakmp` command:

```
hostname(config)## isakmp keepalive threshold <number> retry <number>
hostname(config-tunnel-ipsec)#
```

For example, the following command sets the ISAKMP keepalive threshold to 15 seconds and sets the retry interval to 10 seconds:

```
hostname(config-tunnel-ipsec)# isakmp keepalive threshold 15 retry 10
hostname(config-tunnel-ipsec)#
```

The default value for the `threshold` parameter for LAN-to-LAN is 10, and the default value for the `retry` parameter is 2.

To specify that the central site (“head end”) should never initiate ISAKMP monitoring, enter the following command:

```
hostname(config-tunnel-ipsec)# isakmp keepalive threshold infinite
hostname(config-tunnel-ipsec)#
```

**Step 7** Specify the ISAKMP hybrid authentication method, XAUTH or hybrid XAUTH.

You use `isakmp ikev1-user-authentication` command to implement hybrid XAUTH authentication when you need to use digital certificates for ASA 1000V authentication and a different, legacy method for remote VPN user authentication, such as RADIUS, TACACS+ or SecurID. Hybrid XAUTH breaks phase 1 of IKE down into the following two steps, together called hybrid authentication:

a. The ASA 1000V authenticates to the remote VPN user with standard public key techniques. This establishes an IKE security association that is unidirectionally authenticated.

b. An XAUTH exchange then authenticates the remote VPN user. This extended authentication can use one of the supported legacy authentication methods.
Before the authentication type can be set to hybrid, you must configure the authentication server, create a preshared key, and configure a trustpoint.

For example, the following commands enable hybrid XAUTH for a connection profile called example-group:

```plaintext
hostname(config)# tunnel-group example-group type remote-access
hostname(config)# tunnel-group example-group ipsec-attributes
hostname(config-tunnel-ipsec)# isakmp ikev1-user-authentication hybrid
hostname(config-tunnel-ipsec)#
```

## Group Policies

This section describes group policies and how to configure them. It includes the following sections:

- Default Group Policy, page 27-7
- Configuring Group Policies, page 27-9

A group policy is a set of user-oriented attribute/value pairs for IPsec connections that are stored either internally (locally) on the device or externally on a RADIUS server. The connection profile uses a group policy that sets terms for user connections after the tunnel is established. Group policies let you apply whole sets of attributes to a user or a group of users, rather than having to specify each attribute individually for each user.

Enter the `group-policy` commands in global configuration mode to assign a group policy to users or to modify a group policy for specific users.

The ASA 1000V includes a default group policy. In addition to the default group policy, which you can modify but not delete, you can create one or more group policies specific to your environment.

You can configure internal and external group policies. Internal groups are configured on the ASA 1000V’s internal database. External groups are configured on an external authentication server, such as RADIUS. Group policies include the following attributes:

- Identity
- Server definitions
- Tunneling protocols
- IPsec settings
- Filters
- Connection settings

### Default Group Policy

The ASA 1000V supplies a default group policy. You can modify this default group policy, but you cannot delete it. A default group policy, named DfltGrpPolicy, always exists on the ASA 1000V, but this default group policy does not take effect unless you configure the ASA 1000V to use it. When you configure other group policies, any attribute that you do not explicitly specify takes its value from the default group policy. To view the default group policy, enter the following command:
hostname(config)# show running-config all group-policy DfltGrpPolicy
hostname(config)#

To configure the default group policy, enter the following command:

hostname(config)# group-policy DfltGrpPolicy internal
hostname(config)#

**Note**
The default group policy is always internal. Despite the fact that the command syntax is
hostname(config)# group-policy DfltGrpPolicy {internal | external}, you cannot change its type to external.

To change any of the attributes of the default group policy, use the `group-policy attributes` command to enter attributes mode, then specify the commands to change whatever attributes that you want to modify:

hostname(config)# group-policy DfltGrpPolicy attributes

**Note**
The attributes mode applies only to internal group policies.

The default group policy, DfltGrpPolicy, that the ASA 1000V provides is as follows:

```
show runn all group-policy DfltGrpPolicy
group-policy DfltGrpPolicy internal
group-policy DfltGrpPolicy attributes
  banner none
  wins-server none
dns-server none
dhcp-network-scope none
vpn-access-hours none
vpn-simultaneous-logins 3
vpn-idle-timeout 30
vpn-idle-timeout alert-interval 1
vpn-session-timeout none
vpn-session-timeout alert-interval 1
vpn-filter none
ipv6-vpn-filter none
vpn-tunnel-protocolikev1ikev2l2tp-ipsec
password-storage disable
ip-comp disable
re-xauth disable
group-lock none
pfs disable
ipsec-udp disable
ipsec-udp-port 10000
split-tunnel-policy tunnelall
split-tunnel-network-list none
default-domain none
split-dns none
split-tunnel-all-dns disable
intercept-dhcp 255.255.255.255 disable
secure-unit-authentication disable
user-authentication disable
user-authentication-idle-timeout 30
ip-phone-bypass disable
leap-bypass disable
nem disable
backup-servers keep-client-config
msie-proxy server none
```
You can modify the default group policy, and you can also create one or more group policies specific to your environment.

Configuring Group Policies

A group policy can apply to any kind of tunnel. In each case, if you do not explicitly define a parameter, the group takes the value from the default group policy. To configure a group policy, follow the steps in the subsequent sections.

Configuring an External Group Policy

External group policies take their attribute values from the external server that you specify. For an external group policy, you must identify the AAA server group that the ASA 1000V can query for attributes and specify the password to use when retrieving attributes from the external AAA server group. If you are using an external authentication server, and if your external group-policy attributes exist in the same RADIUS server as the users that you plan to authenticate, you have to make sure that there is no name duplication between them.

Note: External group names on the ASA 1000V refer to user names on the RADIUS server. In other words, if you configure external group X on the ASA 1000V, the RADIUS server sees the query as an authentication request for user X. So external groups are really just user accounts on the RADIUS server that have special meaning to the ASA 1000V. If your external group attributes exist in the same RADIUS server as the users that you plan to authenticate, there must be no name duplication between them.

The ASA 1000V supports user authorization on an external LDAP or RADIUS server. Before you configure the ASA 1000V to use an external server, you must configure the server with the correct ASA 1000V authorization attributes and, from a subset of these attributes, assign specific permissions to individual users.

To configure an external group policy, do the following steps specify a name and type for the group policy, along with the server-group name and a password:

```
hostname(config)# group-policy group_policy_name type server-group server_group_name password server_password
hostname(config)#
```

Note: For an external group policy, RADIUS is the only supported AAA server type.
For example, the following command creates an external group policy named ExtGroup that gets its attributes from an external RADIUS server named ExtRAD and specifies that the password to use when retrieving the attributes is newpassword:

```
hostname(config)# group-policy ExtGroup external server-group ExtRAD password newpassword
hostname(config)#
```

You can configure several vendor-specific attributes (VSAs). If a RADIUS server is configured to return the Class attribute (#25), the ASA 1000V uses that attribute to authenticate the Group Name. On the RADIUS server, the attribute must be formatted as: OU=groupname; where groupname is identical to the Group Name configured on the ASA 1000V—for example, OU=Finance.

**Configuring an Internal Group Policy**

To configure an internal group policy, specify a name and type for the group policy:

```
hostname(config)# group-policy group_policy_name type
hostname(config)#
```

For example, the following command creates the internal group policy named GroupPolicy1:

```
hostname(config)# group-policy GroupPolicy1 internal
hostname(config)#
```

The default type is `internal`.

You can initialize the attributes of an internal group policy to the values of a preexisting group policy by appending the keyword `from` and specifying the name of the existing policy:

```
hostname(config)# group-policy group_policy_name internal from group_policy_name
hostname(config-group-policy)#
hostname(config-group-policy)#
```

**Configuring Group Policy Attributes**

For internal group policies, you can specify particular attribute values. To begin, enter group-policy attributes mode, by entering the `group-policy attributes` command in global configuration mode.

```
hostname(config)# group-policy name attributes
hostname(config-group-policy)#
```

The prompt changes to indicate the mode change. The `group-policy-attributes` mode lets you configure attribute-value pairs for a specified group policy. In `group-policy-attributes` mode, explicitly configure the attribute-value pairs that you do not want to inherit from the default group. The commands to do this are described in the following sections.

**Configuring Site-to-Site VPN-Specific Attributes**

Follow the steps in this section to set the VPN attribute values. The VPN attributes defines a Tunneling protocol for the group policy, an ACL to apply to the site-to-site VPN session, and an idle timeout for the connection.
Step 1  Specify the VPN tunnel type for this group policy.

    vpn-tunnel-protocol {ikev1 | ikev2}

The default is IPsec. To remove the attribute from the running configuration, enter the no form of this command.

The parameter values for this command follow:

- **ikev1**—Negotiates an IPsec IKEv1 tunnel between two peers (the Cisco VPN Client or another secure gateway). Creates security associations that govern authentication, encryption, encapsulation, and key management.

- **ikev2**—Negotiates an IPsec IKEv2 tunnel between two peers (the AnyConnect Secure Mobility Client or another secure gateway). Creates security associations that govern authentication, encryption, encapsulation, and key management.

Enter this command to configure one or more tunneling modes. You must configure at least one tunneling mode for users to connect over a VPN tunnel.

The following example shows how to configure the IPsec IKEv1 tunneling mode for the group policy named FirstGroup:

```
hostname(config)# group-policy FirstGroup attributes
hostname(config-group-policy)# vpn-tunnel-protocol ikev1
```

Step 2  Specify the name of the ACL to apply to VPN session, using the `vpn-filter` command in group policy mode.

```
hostname(config-group-policy)# vpn-filter {value ACL name | none}
```

You configure ACLs to permit or deny various types of traffic for this group policy. You then enter the `vpn-filter` command to apply those ACLs.

To remove the ACL, including a null value created by entering the `vpn-filter none` command, enter the no form of this command. The no option allows inheritance of a value from another group policy.

A group policy can inherit this value from another group policy. To prevent inheriting a value, enter the `none` keyword instead of specifying an ACL name. The `none` keyword indicates that there is no access list and sets a null value, thereby disallowing an access list.

The following example shows how to set a filter that invokes an access list named acl_vpn for the group policy named FirstGroup:

```
hostname(config)# group-policy FirstGroup attributes
hostname(config-group-policy)# vpn-filter acl_vpn
```

A `vpn-filter` command is applied to post-decrypted traffic after it exits a tunnel and pre-encrypted traffic before it enters a tunnel. An ACL that is used for a vpn-filter should NOT also be used for an interface access-group. When a `vpn-filter` command is applied to a group policy that governs Remote Access VPN client connections, the ACL should be configured with the client assigned IP addresses in the `src_ip` position of the ACL and the local network in the `dest_ip` position of the ACL.

When a `vpn-filter` command is applied to a group-policy that governs a LAN to LAN VPN connection, the ACL should be configured with the remote network in the `src_ip` position of the ACL and the local network in the `dest_ip` position of the ACL.
Caution should be used when constructing the ACLs for use with the vpn-filter feature. The ACLs are constructed with the post-decrypted traffic in mind. However, ACLs are also applied to the traffic in the opposite direction. For this pre-encrypted traffic that is destined for the tunnel, the ACLs are constructed with the src_ip and dest_ip positions swapped.

In the following example, the vpn-filter is used with a Remote Access VPN client. This example assumes that the client assigned IP address is 10.10.10.1/24 and the local network is 192.168.1.0/24.

The following ACE will allow the Remote Access VPN client to telnet to the local network:

```plaintext
hostname(config-group-policy)# access-list vpnfilt-ra permit 10.10.10.1 255.255.255.255 192.168.1.0 255.255.255.0 eq 23
```

The following ACE will allow the local network to telnet to the Remote Access client:

```plaintext
hostname(config-group-policy)# access-list vpnfilt-ra permit 10.10.10.1 255.255.255.255 eq 23 192.168.1.0 255.255.255.0
```

Note: The ACE access-list vpnfilt-ra permit 10.10.10.1 255.255.255.255 192.168.1.0 255.255.255.0 eq 23 will allow the local network to initiate a connection to the Remote Access client on any TCP port if it uses a source port of 23. The ACE access-list vpnfilt-ra permit 10.10.10.1 255.255.255.0 eq 23 192.168.1.0 255.255.255.0 will allow the Remote Access client to initiate a connection to the local network on any TCP port if it uses a source port of 23.

In the next example, the vpn-filter is used with a LAN to LAN VPN connection. This example assumes that the remote network is 10.0.0.0/24 and the local network is 192.168.1.0/24.

The following ACE will allow the remote network to telnet to the local network:

```plaintext
hostname(config-group-policy)# access-list vpnfilt-l2l permit 10.0.0.0 255.255.255.0 192.168.1.0 255.255.255.0 eq 23
```

The following ACE will allow the local network to telnet to the remote network:

```plaintext
hostname(config-group-policy)# access-list vpnfilt-l2l permit 10.0.0.0 255.255.255.0 eq 23 192.168.1.0 255.255.255.0
```

Note: The ACE access-list vpnfilt-l2l permit 10.0.0.0 255.255.255.0 192.168.1.0 255.255.255.0 eq 23 will allow the local network to initiate a connection to the remote network on any TCP port if it uses a source port of 23. The ACE access-list vpnfilt-l2l permit 10.0.0.0 255.255.255.0 eq 23 192.168.1.0 255.255.255.0 will allow the remote network to initiate a connection to the local network on any TCP port if it uses a source port of 23.

**Step 3** Configure the idle timeout period by entering the `vpn-idle-timeout` command in group-policy configuration mode or in username configuration mode:

```plaintext
hostname(config-group-policy)# vpn-idle-timeout {minutes | none}
hostname(config-group-policy)#
```

The minimum time is 1 minute, and the maximum time is 35791394 minutes. The default is 30 minutes. If there is no communication activity on the connection in this period, the ASA 1000V terminates the connection.
A group policy can inherit this value from another group policy. To prevent inheriting a value, enter the none keyword instead of specifying a number of minutes with this command. The none keyword also permits an unlimited idle timeout period. It sets the idle timeout to a null value, thereby disallowing an idle timeout.

The following example shows how to set a VPN idle timeout of 15 minutes for the group policy named FirstGroup:

```
hostname(config)# group-policy FirstGroup attributes
hostname(config-group-policy)# vpn-idle-timeout 15
```

**Step 4** Configure the time at which an idle-timeout alert message is displayed to the user using the `vpn-idle-timeout alert-interval {minutes | none}` command. This alert message tells users how many minutes left they have until their VPN session is disconnected due to inactivity.

The following example shows how to set `vpn-idle-timeout alert-interval` so that users will be notified 20 minutes before their VPN session is disconnected due to inactivity. You can specify a range of 1-30 minutes.

```
hostname(config-webvpn)# vpn-idle-timeout alert-interval 20
```

The `none` parameter of the command indicates that users will not receive an alert. The `no` form of the command: `no vpn-idle-timeout alert-interval` indicates that the VPN idle timeout alert-interval attribute will be inherited from the Default Group Policy.

---

### Configuring Domain Attributes for Tunneling

You can specify a default domain name for tunneled packets or a list of domains to be resolved through the split tunnel. The following sections describe how to set these domains.

#### Defining a Default Domain Name for Tunneled Packets

The ASA 1000V passes the default domain name to the IPsec client to append to DNS queries that omit the domain field. When there are no default domain names, users inherit the default domain name in the default group policy. To specify the default domain name for users of the group policy, enter the `default-domain` command in group-policy configuration mode. To delete a domain name, enter the `no` form of this command.

```
hostname(config-group-policy)# default-domain {value domain-name | none}
hostname(config-group-policy)# no default-domain [domain-name]
```

The `value domain-name` parameter identifies the default domain name for the group. To specify that there is no default domain name, enter the `none` keyword. This command sets a default domain name with a null value, which disallows a default domain name and prevents inheriting a default domain name from a default or specified group policy.

To delete all default domain names, enter the `no default-domain` command without arguments. This command deletes all configured default domain names, including a null list if you created one by entering the `default-domain` command with the `none` keyword. The `no` form allows inheriting a domain name.

The following example shows how to set a default domain name of FirstDomain for the group policy named FirstGroup:

```
hostname(config)# group-policy FirstGroup attributes
```
### Defining a List of Domains for Split Tunneling

Enter a list of domains to be resolved through the split tunnel. Enter the `split-dns` command in group-policy configuration mode. To delete a list, enter the `no` form of this command.

When there are no split tunneling domain lists, users inherit any that exist in the default group policy. To prevent users from inheriting such split tunneling domain lists, enter the `split-dns` command with the `none` keyword.

To delete all split tunneling domain lists, enter the `no` `split-dns` command without arguments. This deletes all configured split tunneling domain lists, including a null list created by issuing the `split-dns` command with the `none` keyword.

The parameter `value domain-name` provides a domain name that the ASA 1000V resolves through the split tunnel. The `none` keyword indicates that there is no split DNS list. It also sets a split DNS list with a null value, thereby disallowing a split DNS list, and prevents inheriting a split DNS list from a default or specified group policy. The syntax of the command is as follows:

```
hostname(config-group-policy)# split-dns {value domain-name1 [domain-name2... domain-nameN] | none}
```

Enter a single space to separate each entry in the list of domains. There is no limit on the number of entries, but the entire string can be no longer than 255 characters. You can use only alphanumeric characters, hyphens (-), and periods (.). If the default domain name is to be resolved through the tunnel, you must explicitly include that name in this list.

The following example shows how to configure the domains Domain1, Domain2, Domain3, and Domain4 to be resolved through split tunneling for the group policy named FirstGroup:

```
hostname(config)# group-policy FirstGroup attributes
hostname(config-group-policy)# split-dns value Domain1 Domain2 Domain3 Domain4
```

### Configuring DHCP Intercept

A Microsoft XP anomaly results in the corruption of domain names if split tunnel options exceed 255 bytes. To avoid this problem, the ASA 1000V limits the number of routes it sends to 27 to 40 routes, with the number of routes dependent on the classes of the routes.

DHCP Intercept lets Microsoft Windows XP clients use split-tunneling with the ASA 1000V. The ASA 1000V replies directly to the Microsoft Windows XP client DHCP Inform message, providing that client with the subnet mask, domain name, and classless static routes for the tunnel IP address. For Windows clients prior to Windows XP, DHCP Intercept provides the domain name and subnet mask. This is useful in environments in which using a DHCP server is not advantageous.

The `intercept-dhcp` command enables or disables DHCP intercept. The syntax of this command is as follows:

```
[no] intercept-dhcp
```

```
hostname(config-group-policy)# intercept-dhcp netmask (enable | disable)
hostname(config-group-policy)#
```

The `netmask` variable provides the subnet mask for the tunnel IP address. The `no` version of the command removes the DHCP intercept from the configuration.

The following example shows how to set DHCP Intercepts for the group policy named FirstGroup:

```
hostname(config-group-policy)#
```
Configuring Backup Server Attributes

Configure backup servers if you plan on using them. IPsec backup servers let a VPN client connect to the central site when the primary ASA 1000V is unavailable. When you configure backup servers, the ASA 1000V pushes the server list to the client as the IPsec tunnel is established. Backup servers do not exist until you configure them, either on the client or on the primary ASA 1000V.

Configure backup servers either on the client or on the primary ASA 1000V. If you configure backup servers on the ASA 1000V, it pushes the backup server policy to the clients in the group, replacing the backup server list on the client if one is configured.

Note

If you are using hostnames, it is wise to have backup DNS and WINS servers on a separate network from that of the primary DNS and WINS servers. Otherwise, if clients behind a hardware client obtain DNS and WINS information from the hardware client via DHCP, and the connection to the primary server is lost, and the backup servers have different DNS and WINS information, clients cannot be updated until the DHCP lease expires. In addition, if you use hostnames and the DNS server is unavailable, significant delays can occur.

To configure backup servers, enter the `backup-servers` command in group-policy configuration mode:

```
hostname(config-group-policy)# backup-servers {server1 server2... server10 |
clear-client-config | keep-client-config}
```

To remove a backup server, enter the `no` form of this command with the backup server specified. To remove the backup-servers attribute from the running configuration and enable inheritance of a value for backup-servers from another group policy, enter the `no` form of this command without arguments.

```
hostname(config-group-policy)# no backup-servers {server1 server2... server10 |
clear-client-config | keep-client-config}
```

The `clear-client-config` keyword specifies that the client uses no backup servers. The ASA 1000V pushes a null server list.

The `keep-client-config` keyword specifies that the ASA 1000V sends no backup server information to the client. The client uses its own backup server list, if configured. This is the default.

The `server1 server2... server10` parameter list is a space-delimited, priority-ordered list of servers for the VPN client to use when the primary ASA 1000V is unavailable. This list identifies servers by IP address or hostname. The list can be 500 characters long, and it can contain up to 10 entries.

The following example shows how to configure backup servers with IP addresses 10.10.10.1 and 192.168.10.14, for the group policy named FirstGroup:

```
hostname(config)# group-policy FirstGroup attributes
hostname(config-group-policy)# backup-servers 10.10.10.1 192.168.10.14
```
Configuring LAN-to-LAN IPsec VPNs

A LAN-to-LAN VPN connects networks in different geographic locations.
The ASA 1000V supports LAN-to-LAN VPN connections to Cisco or third-party peers when the two peers have IPv4 inside and outside networks (IPv4 addresses on the inside and outside interfaces).
For LAN-to-LAN connections using mixed IPv4 and IPv6 addressing, or all IPv6 addressing, the security appliance supports VPN tunnels if both peers are Cisco ASA 5500 series security appliances, and if both inside networks have matching addressing schemes (both IPv4 or both IPv6).
Specifically, the following topologies are supported when both peers are Cisco ASA 5500 series ASA 1000Vs:

- The ASA 1000Vs have IPv4 inside networks and the outside network is IPv6 (IPv4 addresses on the inside interfaces and IPv6 addresses on the outside interfaces).
- The ASA 1000Vs have IPv6 inside networks and the outside network is IPv4 (IPv6 addresses on the inside interface and IPv4 addresses on the outside interfaces).
- The ASA 1000Vs have IPv6 inside networks and the outside network is IPv6 (IPv6 addresses on the inside and outside interfaces).

**Note**
The ASA supports LAN-to-LAN IPsec connections with Cisco peers, and with third-party peers that comply with all relevant standards.

This chapter describes how to build a LAN-to-LAN VPN connection. It includes the following sections:

- Summary of the Configuration, page 28-2
- Configuring Interfaces, page 28-2
- Configuring ISAKMP Policy and Enabling ISAKMP on the Outside Interface, page 28-3
- Creating an IKEv1 Transform Set, page 28-5
- Creating an IKEv2 Proposal, page 28-6
- Configuring an ACL, page 28-7
- Defining a Tunnel Group, page 28-8
- Creating a Crypto Map and Applying It To an Interface, page 28-9
Summary of the Configuration

This section provides a summary of the example LAN-to-LAN configuration described in this chapter. Later sections provide step-by-step instructions.

hostname(config)# interface ethernet0/0
hostname(config-if)# ip address 10.10.4.100 255.255.0.0
hostname(config-if)# nameif outside
hostname(config-if)# no shutdown
hostname(config)# crypto ikev1 policy 1
hostname(config-ikev1-policy)# authentication pre-share
hostname(config-ikev1-policy)# encryption 3des
hostname(config-ikev1-policy)# hash sha
hostname(config-ikev1-policy)# group 2
hostname(config-ikev1-policy)# lifetime 43200
hostname(config)# crypto ikev1 enable outside
hostname(config)# crypto ikev2 policy 1
hostname(config-ikev2-policy)# encryption 3des
hostname(config-ikev2-policy)# group 2
hostname(config-ikev2-policy)# prf sha
hostname(config-ikev2-policy)# lifetime 43200
hostname(config)# crypto ikev2 enable outside
hostname(config)# crypto ipsec ikev1 transform-set FirstSet esp-3des esp-md5-hmac
hostname(config)# crypto ipsec ikev2 ipsec-proposal secure
hostname(config)# crypto ipsec-proposal # protocol esp encryption 3des aes des
hostname(config)# access-list l2l_list extended permit ip 192.168.0.0 255.255.0.0 150.150.0.0 255.255.0.0
hostname(config)# tunnel-group 10.10.4.108 type ipsec-121
hostname(config)# tunnel-group 10.10.4.108 ipsec-attributes
hostname(config)# crypto ikev1 pre-shared-key 44kkoal5963jnfx
hostname(config)# crypto map abcmap 1 match address l2l_list
hostname(config)# crypto map abcmap 1 set peer 10.10.4.108
hostname(config)# crypto map abcmap 1 set ikev1 transform-set FirstSet
hostname(config)# crypto map abcmap 1 set ikev2 ipsec-proposal secure
hostname(config)# crypto map abcmap interface outside
hostname(config)# write memory

Configuring Interfaces

An ASA 1000V has at least two interfaces, referred to here as outside and inside. Typically, the outside interface is connected to the public Internet, while the inside interface is connected to a private network and is protected from public access.

To begin, configure and enable two interfaces on the ASA 1000V. Then, assign a name, IP address and subnet mask. Optionally, configure its security level, speed, and duplex operation on the security appliance.

To configure interfaces, perform the following steps, using the command syntax in the examples:

---

**Step 1** To enter Interface configuration mode, in global configuration mode enter the `interface` command with the default name of the interface to configure. In the following example the interface is ethernet0.

hostname(config)# interface ethernet0/0
hostname(config-if)#

**Step 2** To set the IP address and subnet mask for the interface, enter the `ip address` command. In the following example the IP address is 10.10.4.100 and the subnet mask is 255.255.0.0.
Step 3 To name the interface, enter the `nameif` command, maximum of 48 characters. You cannot change this name after you set it. In the following example the name of the ethernet0 interface is outside.

```
hostname(config-if)# nameif outside
```

Step 4 To enable the interface, enter the `no` version of the `shutdown` command. By default, interfaces are disabled.

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

Step 5 To save your changes, enter the `write memory` command.

```
hostname(config-if)# write memory
```

Step 6 To configure a second interface, use the same procedure.

---

**Configuring ISAKMP Policy and Enabling ISAKMP on the Outside Interface**

ISAKMP is the negotiation protocol that lets two hosts agree on how to build an IPsec security association (SA). It provides a common framework for agreeing on the format of SA attributes. This includes negotiating with the peer about the SA, and modifying or deleting the SA. ISAKMP separates negotiation into two phases: Phase 1 and Phase 2. Phase 1 creates the first tunnel, which protects later ISAKMP negotiation messages. Phase 2 creates the tunnel that protects data.

IKE uses ISAKMP to setup the SA for IPsec to use. IKE creates the cryptographic keys used to authenticate peers.

The ASA 1000V supports IKEv1 for connections from the legacy Cisco VPN client, and IKEv2 for the AnyConnect VPN client.

To set the terms of the ISAKMP negotiations, you create an IKE policy, which includes the following:

- The authentication type required of the IKEv1 peer, either RSA signature using certificates or preshared key (PSK).
- An encryption method, to protect the data and ensure privacy.
- A Hashed Message Authentication Codes (HMAC) method to ensure the identity of the sender, and to ensure that the message has not been modified in transit.
- A Diffie-Hellman group to determine the strength of the encryption-key-determination algorithm. The ASA 1000V uses this algorithm to derive the encryption and hash keys.
- For IKEv2, a separate pseudo-random function (PRF) used as the algorithm to derive keying material and hashing operations required for the IKEv2 tunnel encryption, etc.
- A limit to the time the ASA 1000V uses an encryption key before replacing it.
With IKEv1 policies, for each parameter, you set one value. For IKEv2, you can configure multiple encryption and authentication types, and multiple integrity algorithms for a single policy. The ASA 1000V orders the settings from the most secure to the least secure and negotiates with the peer using that order. This allows you to potentially send a single proposal to convey all the allowed transforms instead of the need to send each allowed combination as with IKEv1.

The following sections provide procedures for creating IKEv1 and IKEv2 policies and enabling them on an interface:

- Configuring ISAKMP Policies for IKEv1 Connections, page 28-4
- Configuring ISAKMP Policies for IKEv2 Connections, page 28-5

### Configuring ISAKMP Policies for IKEv1 Connections

To configure ISAKMP policies for IKEv1 connections, use the `crypto ikev1 policy` command to enter IKEv1 policy configuration mode where you can configure the IKEv1 parameters:

```plaintext
crypto ikev1 policy priority
```

Perform the following steps and use the command syntax in the following examples as a guide.

---

**Step 1** Enter IPsec IKEv1 policy configuration mode. For example:

```plaintext
hostname(config)# crypto ikev1 policy 1
hostname(config-ikev1-policy)#
```

**Step 2** Set the authentication method. The following example configures a preshared key:

```plaintext
hostname(config-ikev1-policy)# authentication pre-share
hostname(config-ikev1-policy)#
```

**Step 3** Set the encryption method. The following example configures 3DES:

```plaintext
hostname(config-ikev1-policy)# encryption 3des
hostname(config-ikev1-policy)#
```

**Step 4** Set the HMAC method. The following example configures SHA-1:

```plaintext
hostname(config-ikev1-policy)# hash sha
hostname(config-ikev1-policy)#
```

**Step 5** Set the Diffie-Hellman group. The following example configures Group 2:

```plaintext
hostname(config-ikev1-policy)# group 2
hostname(config-ikev1-policy)#
```

**Step 6** Set the encryption key lifetime. The following example configures 43,200 seconds (12 hours):

```plaintext
hostname(config-ikev1-policy)# lifetime 43200
hostname(config-ikev1-policy)#
```

**Step 7** Enable IKEv1 on the interface named outside:

```plaintext
hostname(config)# crypto ikev1 enable outside
hostname(config)#
```

**Step 8** To save your changes, enter the `write memory` command:

```plaintext
hostname(config)# write memory
hostname(config)#
```
Creating ISAKMP Policies for IKEv2 Connections

To configure ISAKMP policies for IKEv2 connections, use the `crypto ikev2 policy` command to enter IKEv2 policy configuration mode where you can configure the IKEv2 parameters:

```
crypto ikev2 policy priority
```

Perform the following steps and use the command syntax in the following examples as a guide:

**Step 1** Enter IPsec IKEv2 policy configuration mode. For example:
```
hostname(config)# crypto ikev2 policy 1
hostname(config-ikev2-policy)#
```

**Step 2** Set the encryption method. The following example configures 3DES:
```
hostname(config-ikev2-policy)# encryption 3des
hostname(config-ikev2-policy)#
```

**Step 3** Set the Diffie-Hellman group. The following example configures Group 2:
```
hostname(config-ikev2-policy)# group 2
hostname(config-ikev2-policy)#
```

**Step 4** Set the pseudo-random function (PRF) used as the algorithm to derive keying material and hashing operations required for the IKEv2 tunnel encryption. The following example configures SHA-1 (an HMAC variant):
```
hostname(config-ikev2-policy)# prf sha
hostname(config-ikev2-policy)#
```

**Step 5** Set the encryption key lifetime. The following example configures 43,200 seconds (12 hours):
```
hostname(config-ikev2-policy)# lifetime 43200
hostname(config-ikev2-policy)#
```

**Step 6** Enable IKEv2 on the interface named outside:
```
hostname(config)# crypto ikev2 enable outside
hostname(config)#
```

**Step 7** To save your changes, enter the `write memory` command:
```
hostname(config)# write memory
hostname(config)#
```

Creating an IKEv1 Transform Set

An IKEv1 transform set combines an encryption method and an authentication method. During the IPsec security association negotiation with ISAKMP, the peers agree to use a particular transform set to protect a particular data flow. The transform set must be the same for both peers.

A transform set protects the data flows for the access list specified in the associated crypto map entry. You can create transform sets in the ASA 1000V configuration, and then specify a maximum of 11 of them in a crypto map or dynamic crypto map entry.
Creating an IKEv2 Proposal

For IKEv2, you can configure multiple encryption and authentication types, and multiple integrity algorithms for a single policy. The ASA 1000V orders the settings from the most secure to the least secure and negotiates with the peer using that order. This allows you to potentially send a single proposal to convey all the allowed transforms instead of the need to send each allowed combination as with IKEv1.

Table 28-1 lists valid IKEv2 encryption and authentication methods.

### Table 28-1 Valid IKEv2 Encryption and Integrity Methods

<table>
<thead>
<tr>
<th>Valid Encryption Methods</th>
<th>Valid Integrity Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>des</td>
<td>sha (default)</td>
</tr>
<tr>
<td>3des (default)</td>
<td>md5</td>
</tr>
<tr>
<td>aes</td>
<td></td>
</tr>
<tr>
<td>aes-192</td>
<td></td>
</tr>
<tr>
<td>aes-256</td>
<td></td>
</tr>
</tbody>
</table>
To configure an IKEv2 proposal, perform the following steps:

**Step 1** In global configuration mode, use the `crypto ipsec ikev2 ipsec-proposal` command to enter ipsec proposal configuration mode where you can specify multiple encryption and integrity types for the proposal. In this example, `secure` is the name of the proposal:

```
hostname(config)# crypto ipsec ikev2 ipsec-proposal secure
```

**Step 2** Then enter a protocol and encryption types. ESP is the only supported protocol. For example:

```
hostname(config-ipsec-proposal)# protocol esp encryption 3des aes des
```

**Step 3** Enter an integrity type. For example:

```
hostname(config-ipsec-proposal)# protocol esp integrity sha-1
```

**Step 4** Save your changes.

---

**Configuring an ACL**

The adaptive security appliance uses access control lists to control network access. By default, the adaptive security appliance denies all traffic. You need to configure an ACL that permits traffic. For more information, see Chapter 9, “Information About Access Lists.”

The ACLs that you configure for this LAN-to-LAN VPN control connections are based on the source and translated destination IP addresses. Configure ACLs that mirror each other on both sides of the connection.

An ACL for VPN traffic uses the translated address. For more information, see the “IP Addresses Used for Access Lists When You Use NAT” section on page 9-2.

To configure an ACL, perform the following steps:

**Step 1** Enter the `access-list extended` command. The following example configures an ACL named `l2l_list` that lets traffic from IP addresses in the 192.168.0.0 network travel to the 150.150.0.0 network. The syntax is `access-list listname extended permit ip source-ipaddress source-netmask destination-ipaddress destination-netmask`:

```
hostname(config)# access-list l2l_list extended permit ip 192.168.0.0 255.255.0.0 150.150.0.0 255.255.0.0
```

**Step 2** Configure an ACL for the ASA 1000V on the other side of the connection that mirrors the ACL above. In the following example the prompt for the peer is `hostname2`.

```
hostname2(config)# access-list l2l_list extended permit ip 150.150.0.0 255.255.0.0 192.168.0.0 255.255.0.0
```

**Note** For more information on configuring an ACL with a vpn-filter, see “Configuring Site-to-Site VPN-Specific Attributes” section on page 27-10.
Defining a Tunnel Group

A tunnel group is a set of records that contain tunnel connection policies. You configure a tunnel group to identify AAA servers, specify connection parameters, and define a default group policy. The ASA 1000V stores tunnel groups internally.

There are two default tunnel groups in the ASA 1000V: DefaultRAGroup, which is the default IPsec remote-access tunnel group, and DefaultL2Lgroup, which is the default IPsec LAN-to-LAN tunnel group. You can modify them but not delete them.

You can also create one or more new tunnel groups to suit your environment. The ASA 1000V uses these groups to configure default tunnel parameters for remote access and LAN-to-LAN tunnel groups when there is no specific tunnel group identified during tunnel negotiation.

To establish a basic LAN-to-LAN connection, you must set two attributes for a tunnel group:

- Set the connection type to IPsec LAN-to-LAN.
- Configure an authentication method for the IP, in the following example, preshared key for IKEv1 and IKEv2.

**Note** To use VPNs, including tunnel groups, the ASA must be in single-routed mode. The commands to configure tunnel-group parameters do not appear in any other mode.

**Step 1**

To set the connection type to IPsec LAN-to-LAN, enter the `tunnel-group` command. The syntax is `tunnel-group name type type`, where `name` is the name you assign to the tunnel group, and `type` is the type of tunnel. The tunnel types as you enter them in the CLI are:

- **remote-access** (IPsec, SSL, and clientless SSL remote access)
- **ipsec-l2l** (IPsec LAN to LAN)

In the following example the name of the tunnel group is the IP address of the LAN-to-LAN peer, 10.10.4.108.

```
hostname(config)# tunnel-group 10.10.4.108 type ipsec-l2l
hostname(config)#
```

**Note** LAN-to-LAN tunnel groups that have names that are not an IP address can be used only if the tunnel authentication method is Digital Certificates and/or the peer is configured to use Aggressive Mode.

**Step 2**

To set the authentication method to preshared key, enter the `ipsec-attributes` mode and then enter the `pre-shared-key` command to create the preshared key. You need to use the same preshared key on both ASA 1000V s for this LAN-to-LAN connection.

The key is an alphanumeric string of 1-128 characters.

In the following example the IKEv1 preshared key is 44kkaol59636jnfx:

```
hostname(config)# tunnel-group 10.10.4.108 ipsec-attributes
hostname(config-tunnel-ipsec)# pre-shared-key 44kkaol59636jnfx
```

In the next example, the IKEv2 preshared key is configured also as 44kkaol59636jnfx:

```
hostname(config-tunnel-ipsec)# ikev2 local-authentication pre-shared-key 44kkaol59636jnfx
```

**Step 3**

Save your changes.

```
hostname(config)# write memory
```
Creating a Crypto Map and Applying It To an Interface

Crypto map entries pull together the various elements of IPsec security associations, including the following:

- Which traffic IPsec should protect, which you define in an access list.
- Where to send IPsec-protected traffic, by identifying the peer.
- What IPsec security applies to this traffic, which a transform set specifies.
- The local address for IPsec traffic, which you identify by applying the crypto map to an interface.

For IPsec to succeed, both peers must have crypto map entries with compatible configurations. For two crypto map entries to be compatible, they must, at a minimum, meet the following criteria:

- The crypto map entries must contain compatible crypto access lists (for example, mirror image access lists). If the responding peer uses dynamic crypto maps, the entries in the ASA 1000V crypto access list must be "permitted" by the peer’s crypto access list.
- The crypto map entries each must identify the other peer (unless the responding peer is using a dynamic crypto map).
- The crypto map entries must have at least one transform set in common.

If you create more than one crypto map entry for a given interface, use the sequence number (seq-num) of each entry to rank it: the lower the seq-num, the higher the priority. At the interface that has the crypto map set, the ASA 1000V evaluates traffic against the entries of higher priority maps first.

Create multiple crypto map entries for a given interface if either of the following conditions exist:

- Different peers handle different data flows.
- You want to apply different IPsec security to different types of traffic (to the same or separate peers), for example, if you want traffic between one set of subnets to be authenticated, and traffic between another set of subnets to be both authenticated and encrypted. In this case, define the different types of traffic in two separate access lists, and create a separate crypto map entry for each crypto access list.

To create a crypto map and apply it to the outside interface in global configuration mode, enter several of the `crypto map` commands. These commands use a variety of arguments, but the syntax for all of them begin with `crypto map map-name seq-num`. In the following example the map-name is abcmap, the sequence number is 1.

Enter these commands in global configuration mode:

**Step 1** To assign an access list to a crypto map entry, enter the `crypto map match address` command.

The syntax is `crypto map map-name seq-num match address aclname`. In the following example the map name is abcmap, the sequence number is 1, and the access list name is `121_list`.

```
hostname(config)# crypto map abcmap 1 match address 121_list
hostname(config)#
```

**Step 2** To identify the peer(s) for the IPsec connection, enter the `crypto map set peer` command.

The syntax is `crypto map map-name seq-num set peer {ip_address1 | hostname1} [... ip_address10 | hostname10]`. In the following example the peer name is 10.10.4.108.

```
hostname(config)#
```
Step 3 To specify an IKEv1 transform set for a crypto map entry, enter the **crypto map ikev1 set transform-set** command.

The syntax is `crypto map map-name seq-num ikev1 set transform-set transform-set-name`. In the following example the transform set name is `FirstSet`.

```
hostname(config)# crypto map abcmap 1 set transform-set FirstSet
```

Step 4 To specify an IKEv2 proposal for a crypto map entry, enter the **crypto map ikev2 set ipsec-proposal** command:

The syntax is `crypto map map-name seq-num set ikev2 ipsec-proposal proposal-name`. In the following example the proposal name is `secure`.

```
hostname(config)# crypto map abcmap 1 set ikev2 ipsec-proposal secure
```

### Applying Crypto Maps to Interfaces

You must apply a crypto map set to each interface through which IPsec traffic travels. The ASA 1000V supports IPsec on all interfaces. Applying the crypto map set to an interface instructs the ASA 1000V to evaluate all interface traffic against the crypto map set and to use the specified policy during connection or security association negotiations.

Binding a crypto map to an interface also initializes the runtime data structures, such as the security association database and the security policy database. When you later modify a crypto map in any way, the ASA 1000V automatically applies the changes to the running configuration. It drops any existing connections and reestablishes them after applying the new crypto map.

Step 1 To apply the configured crypto map to the outside interface, enter the **crypto map interface** command. The syntax is `crypto map map-name interface interface-name`.

```
hostname(config)# crypto map abcmap interface outside
```

Step 2 Save your changes.

```
hostname(config)# write memory
```
PART 10

Configuring Logging, SNMP, and Smart Call Home
Configuring Logging

This chapter describes how to configure and manage logs for the ASA 1000V and includes the following sections:

- Information About Logging, page 29-1
- Prerequisites for Logging, page 29-4
- Guidelines and Limitations, page 29-5
- Configuring Logging, page 29-5
- Monitoring the Logs, page 29-18
- Configuration Examples for Logging, page 29-18
- Feature History for Logging, page 29-19

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 ASA 1000V system logs provide you with information for monitoring and troubleshooting the ASA 1000V. 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.
Information About Logging

This section includes the following topics:

- Analyzing Syslog Messages, page 29-2
- Syslog Message Format, page 29-2
- Severity Levels, page 29-3
- Message Classes and Range of Syslog IDs, page 29-3
- Filtering Syslog Messages, page 29-3
- Using Custom Message Lists, page 29-4

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 ASA 1000V security policies. These messages help you spot holes that remain open in your security policies.
- Connections that are denied by ASA 1000V 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 ASA 1000V.
- 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>ASA</td>
<td>The syslog message facility code for messages that are generated by the ASA 1000V. 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 29-1 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 29-1 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>

The ASA 1000V 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 ASA 1000V.

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 ASA 1000V 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 ASA 1000V 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 ASA 1000V)

You customize these criteria by creating a message list that you can specify when you set the output destination. Alternatively, you can configure the ASA 1000V 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 ASA 1000V. For example, the snmp class denotes the SNMP agent.

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 212 are associated with the snmp class. Syslog messages associated with the SNMP feature range from 212001 to 212012.

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 IPsec 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.

**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 ASA 1000V, you must specify a logging output destination. If you enable logging without specifying a logging output destination, the ASA 1000V 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 describes the guidelines and limitations of this feature.

- Sending syslogs over TCP is not supported on a standby ASA 1000V.
- The ASA 1000V 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 29-5
- Configuring an Output Destination, page 29-6

**Note**

The minimum configuration depends on what you want to do and what your requirements are for handling syslog messages in the ASA 1000V.

**Enabling Logging**

To enable logging, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>logging enable</strong></td>
<td>Enables logging. To disable logging, enter the <code>no logging enable</code> command.</td>
</tr>
</tbody>
</table>

**Example:**

```
hostname(config)# logging enable
```
Configuring Logging

What to Do Next

See the “Configuring an Output Destination” section on page 29-6.

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 29-7
- Sending Syslog Messages to the Internal Log Buffer, page 29-8
- Sending Syslog Messages to an E-mail Address, page 29-9
- Sending Syslog Messages to ASDM, page 29-10
- Sending Syslog Messages to the Console Port, page 29-10
- Sending Syslog Messages to an SNMP Server, page 29-11
- Sending Syslog Messages to a Telnet or SSH Session, page 29-11
- Creating a Custom Event List, page 29-12
- Generating Syslog Messages in EMBLEM Format to a Syslog Server, page 29-13
- Generating Syslog Messages in EMBLEM Format to Other Output Destinations, page 29-13
- Changing the Amount of Internal Flash Memory Available for Logs, page 29-14
- Configuring the Logging Queue, page 29-14
- Sending All Syslog Messages in a Class to a Specified Output Destination, page 29-15
- Enabling Secure Logging, page 29-15
- Including the Device ID in Non-EMBLEM Format Syslog Messages, page 29-16
- Including the Date and Time in Syslog Messages, page 29-16
- Disabling a Syslog Message, page 29-17
- Changing the Severity Level of a Syslog Message, page 29-17
- Limiting the Rate of Syslog Message Generation, page 29-17
## 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>Step</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>`logging host interface_name syslog_ip [tcp[/port]</td>
<td>udp[/port] [format emblem]`</td>
</tr>
<tr>
<td></td>
<td><code>Example:</code></td>
<td><code>hostname(config)# logging host dmz1 192.168.1.5 udp 1026 format emblem</code></td>
</tr>
<tr>
<td>2</td>
<td>`logging trap {severity_level</td>
<td>message_list}`</td>
</tr>
<tr>
<td></td>
<td><code>Example:</code></td>
<td><code>hostname(config)# logging trap errors</code></td>
</tr>
</tbody>
</table>
### 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

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 2**

logging buffer-size `bytes`

- **Example:**
  
  hostname(config)# logging buffer-size 16384

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 3**

Choose one of the following options:

logging flash-bufferwrap

- **Example:**
  
  hostname(config)# logging flash-bufferwrap
Configuring Logging

Sending Syslog Messages to an E-mail Address

To send syslog messages to an e-mail address, perform the following steps:

**Step 1**

**Command:**

logging mail \{severity_level | message_list\}

**Example:**

hostname(config)# logging mail high-priority

**Purpose:**

Specifies which syslog messages should be sent to an e-mail address. When sent by e-mail, a syslog message appears in the subject line of the e-mail message. For this reason, we recommend configuring this option to notify administrators of syslog messages with high severity levels, such as critical, alert, and emergency.

**Step 2**

**Command:**

logging from-address email_address

**Example:**

hostname(config)# logging from-address xxx-001@example.com

**Purpose:**

Specifies the source e-mail address to be used when sending syslog messages to an e-mail address.
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>Step 1</td>
<td>logging asdm {severity_level</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# logging asdm 2</td>
</tr>
<tr>
<td>Step 2</td>
<td>logging asdm-buffer-size num_of_msgs</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# logging asdm-buffer-size 200</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}olars should be sent to the console port.</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# logging console errors</td>
</tr>
</tbody>
</table>
**Sending Syslog Messages to an SNMP Server**

To enable logging to an SNMP server, enter the following command:

```
hostname(config)# logging history [logging_list | level]
```

**Example:**

```
hostname(config)# logging history errors
```

Enables SNMP logging and specifies which messages are to be sent to SNMP servers. To disable SNMP logging, enter the `no logging history` command.

---

**Sending Syslog Messages to a Telnet or SSH Session**

To send syslog messages to a Telnet or SSH session, perform the following steps:

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>`logging monitor {severity_level</td>
<td>message_list}`</td>
</tr>
<tr>
<td></td>
<td>Example: <code>hostname(config)# logging monitor 6</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>
<tr>
<td>2</td>
<td><code>terminal monitor</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Example: <code>hostname(config)# terminal monitor</code></td>
<td></td>
</tr>
</tbody>
</table>

---
# Configuring Logging

## 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>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>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 ASA 1000V sends syslog messages for severity levels 3, 2, and 1.</td>
</tr>
<tr>
<td><code>logging list name (level level [class message_class] \message start_id[-end_id])</code></td>
<td>The <code>name</code> argument specifies the name of the list. The <code>level level</code> keyword and argument pair specify the severity level. The <code>class message_class</code> keyword and argument pair specify a particular message class. The <code>message start_id[-end_id]</code> keyword and argument pair specify an individual syslog message number or a range of numbers.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>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.</td>
</tr>
<tr>
<td>hostname(config)# logging list notif-list level 3</td>
<td>(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:</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>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.</td>
</tr>
<tr>
<td><code>logging list name (level level [class message_class] \message start_id[-end_id])</code></td>
<td>• Syslog message IDs that fall into the range of 104024 to 105999.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>• All syslog messages with the critical severity level or higher (emergency, alert, or critical).</td>
</tr>
<tr>
<td>hostname(config)# logging list notif-list level critical</td>
<td>• All ha class syslog messages with the warning severity level or higher (emergency, alert, critical, error, or warning).</td>
</tr>
<tr>
<td>hostname(config)# logging list notif-list level warning class ha</td>
<td><strong>Note</strong></td>
</tr>
<tr>
<td></td>
<td>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.</td>
</tr>
</tbody>
</table>
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:

**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>
</tbody>
</table>

**Example:**

```
hostname(config)# logging host interface_1 127.0.0.1 udp format emblem
```

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:

**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>
</tbody>
</table>

**Example:**

```
hostname(config)# logging emblem
```
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>&lt;br&gt;<strong>Example:</strong>&lt;br&gt;hostname(config)# logging flash-maximum-allocation 1200</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><strong>logging flash-minimum-free kbytes</strong>&lt;br&gt;<strong>Example:</strong>&lt;br&gt;hostname(config)# logging flash-minimum-free 4000</td>
</tr>
</tbody>
</table>

Configuring the Logging Queue

To configure the logging queue, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>logging queue message_count</strong>&lt;br&gt;<strong>Example:</strong>&lt;br&gt;hostname(config)# logging queue 300</td>
<td>Specifies the number of syslog messages that the ASA 1000V can hold in its queue before sending them to the configured output destination. The ASA 1000V 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. If the logging queue is set to zero, the queue is the maximum configurable size (8192 messages).</td>
</tr>
</tbody>
</table>
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>

Example:

```
hostname(config)# logging host inside 10.0.0.1 TCP/1500 secure
```

Note Secure logging does not support UDP; an error occurs if you try to use this protocol.
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>
<tbody>
<tr>
<td>logging device-id [hostname</td>
<td>ipaddress interface_name</td>
</tr>
</tbody>
</table>

Example:
hostname(config)# logging device-id hostname

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>logging timestamp</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 no logging timestamp 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>no logging message message_number</td>
<td>Prevents the ASA 1000V from generating a particular syslog message. To reenable a disabled syslog message, enter the <code>logging message message_number</code> command (for example, <code>logging message 113019</code>). To reenable logging of all disabled syslog messages, enter the <code>clear config logging disabled</code> 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>logging message 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 <code>no logging message message_ID level current_severity_level</code> command (for example, <code>no logging message 113019 level 5</code>). To reset the severity level of all modified syslog messages to their settings, enter the <code>clear configure logging level</code> 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></td>
<td><strong>Note</strong> The maximum number of syslog messages that are available to view is 1000, which is the default setting. The maximum number of syslog messages that are available to view is 2000.</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 29-2 lists each feature change

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Syslog messages</td>
<td>8.7(1)</td>
<td>We introduced the following syslog messages: 450002 and 771001-771003.</td>
</tr>
</tbody>
</table>
Configuring SNMP

This chapter describes how to configure SNMP to monitor the ASA 1000V and includes the following sections:

- Information About SNMP, page 30-1
- Prerequisites for SNMP, page 30-7
- Guidelines and Limitations, page 30-8
- Configuring SNMP, page 30-8
- Troubleshooting Tips, page 30-13
- Monitoring SNMP, page 30-15
- Configuration Examples for SNMP, page 30-17
- Where to Go Next, page 30-18
- Additional References, page 30-18
- Feature History for SNMP, page 30-20

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 30-2
- Information About MIBs and Traps, page 30-3
- SNMP Object Identifiers, page 30-3
- SNMP Physical Vendor Type Values, page 30-3
- Supported Tables and Objects in MIBs, page 30-4
- Supported Traps (Notifications), page 30-4
- SNMP Version 3, page 30-6

The ASA 1000V 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 ASA 1000V Ethernet interface lets you monitor the ASA 1000V and through network management systems (NMSs),
such as HP OpenView. The ASA 1000V 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 ASA 1000V 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 ASA 1000V. MIBs are a collection of definitions, and the ASA 1000V 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 ASA 1000V 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 ASA 1000V SNMP agent also replies when a management station asks for information.

### Information About SNMP Terminology

Table 30-1 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 ASA 1000V. 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 ASA 1000V.</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 ASA 1000V software.

If needed, you can also download RFCs, standard MIBs, and standard traps from the following locations:


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**

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 30-2 lists the system-level product OIDs.

<table>
<thead>
<tr>
<th>Product Identifier</th>
<th>sysObjectID</th>
<th>Model Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>ciscoASA1000Vsc</td>
<td>ciscoProducts 1613</td>
<td>Cisco Adaptive Security Appliance 1000V Cloud Firewall Security Context</td>
</tr>
<tr>
<td>ciscoASA1000V</td>
<td>ciscoProducts 1614</td>
<td>Cisco Adaptive Security Appliance 1000V Cloud Firewall</td>
</tr>
</tbody>
</table>

### 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 ASA 1000V SNMP agent. You can use this value to identify the type of component (module, power supply, fan, sensors, CPU, and so on). Table 30-3 lists the physical vendor type values for the ASA 1000V.

<table>
<thead>
<tr>
<th>Item</th>
<th>entPhysicalVendorType OID Description</th>
<th>OID</th>
</tr>
</thead>
<tbody>
<tr>
<td>cevChassis 1194</td>
<td>cevChassisASA100V</td>
<td>1.3.6.1.4.1.9.12.3.1.3.1.1194</td>
</tr>
</tbody>
</table>
**Supported Tables and Objects in MIBs**

Table 30-4 lists the supported tables and objects for the specified MIBs.

**Table 30-4  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, cempMemPoolFreeOvrflw, cempMemPoolHCUsed, cempMemPoolFreeOvrflw, cempMemPoolHCFree</td>
</tr>
<tr>
<td>CISCO-L4L7MODULE-RESOURCE-LIMIT-MIB</td>
<td>ciscoL4L7ResourceLimitTable</td>
</tr>
<tr>
<td>DISMAN-EVENT-MIB</td>
<td>mteTriggerTable, mteTriggerThresholdTable, mteObjectsTable, mteEventTable, mteEventNotificationTable</td>
</tr>
<tr>
<td>DISMAN-EXPRESSION-MIB</td>
<td>expExpressionTable, expObjectTable, expValueTable</td>
</tr>
<tr>
<td>NAT-MIB</td>
<td>natAddrMapTable, natAddrMapIndex, natAddrMapName, natAddrMapGlobalAddrType, natAddrMapGlobalAddrFrom, natAddrMapGlobalAddrTo, natAddrMapGlobalPortFrom, natAddrMapGlobalPortTo, natAddrMapProtocol, natAddrMapAddrUsed, natAddrMapRowStatus</td>
</tr>
</tbody>
</table>

**Supported Traps (Notifications)**

Table 30-5 lists the supported traps (notifications) and their associated MIBs.

**Table 30-5  Supported Traps (Notifications)**

<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>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>
### Table 30-5  Supported Traps (Notifications) (continued)

<table>
<thead>
<tr>
<th>Trap Name</th>
<th>Trap Details</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>clogMessageGenerated</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 <strong>snmp-server enable traps syslog</strong> command is used to enable and disable transmission of these traps.</td>
</tr>
<tr>
<td>clrResourceLimitReached</td>
<td>clrResourceLimitValueType, clrResourceLimitMax, clogOriginIDType, clogOriginID</td>
<td>The <strong>snmp-server enable traps connection-limit-reached</strong> command is used to enable transmission of the connection-limit-reached notification.</td>
</tr>
<tr>
<td>coldStart</td>
<td>—</td>
<td>The SNMP agent has started. The <strong>snmp-server enable traps snmp coldstart</strong> command is used to enable and disable transmission of these traps.</td>
</tr>
<tr>
<td>cpmCPURisingThreshold</td>
<td>cpmCPURisingThresholdValue, cpmCPUInterruptMonIntervalValue, cpmCPUInterruptMonIntervalValue, cpmCPURisingThresholdPeriod, cpmProcessTimeCreated, cpmProcExtUtil5SecRev</td>
<td>The <strong>snmp-server enable traps cpu threshold rising</strong> 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>linkDown</td>
<td>ifIndex, ifAdminStatus, ifOperStatus</td>
<td>The linkdown trap for interfaces. The <strong>snmp-server enable traps snmp linkdown</strong> command is used to enable and disable transmission of these traps.</td>
</tr>
<tr>
<td>linkUp</td>
<td>ifIndex, ifAdminStatus, ifOperStatus</td>
<td>The linkup trap for interfaces. The <strong>snmp-server enable traps snmp linkup</strong> command is used to enable and disable transmission of these traps.</td>
</tr>
<tr>
<td>mteTriggerFired</td>
<td>mteHotTrigger, mteHotTargetName, mteHotContextName, mteHotOID, mteHotValue, c.empMemPoolName, c.empMemPoolHCUsed</td>
<td>The <strong>snmp-server enable traps memory-threshold</strong> command is used to enable the memory threshold notification. The mteHotOID is set to c.empMemPoolHCUsed. The c.empMemPoolName and c.empMemPoolHCUsed objects are sent with the other objects.</td>
</tr>
<tr>
<td>mteTriggerFired</td>
<td>mteHotTrigger, mteHotTargetName, mteHotContextName, mteHotOID, mteHotValue, ifHCInOctets, ifHCOutOctets, ifHighSpeed, entPhysicalName</td>
<td>The <strong>snmp-server enable traps interface-threshold</strong> command is used to enable the interface threshold notification. The entPhysicalName objects are sent with the other objects.</td>
</tr>
</tbody>
</table>
Information About SNMP

SNMP Version 3

This section describes SNMP Version 3 and includes the following topics:

- SNMP Version 3 Overview, page 30-6
- Security Models, page 30-6
- SNMP Groups, page 30-7
- SNMP Users, page 30-7
- SNMP Hosts, page 30-7
- Implementation Differences Between the ASA 1000V and the Cisco IOS, page 30-7

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 ASA 1000V 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.

---

Table 30-5 Supported Traps (Notifications) (continued)

<table>
<thead>
<tr>
<th>natPacketDiscard (NAT-MIB)</th>
<th>ifIndex</th>
<th>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.</th>
</tr>
</thead>
<tbody>
<tr>
<td>warmStart (SNMPv2-MIB)</td>
<td>—</td>
<td>The <code>snmp-server enable traps smtp warmstart</code> command is used to enable and disable transmission of these traps.</td>
</tr>
</tbody>
</table>
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 ASA 1000V. 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 ASA 1000V.

Implementation Differences Between the ASA 1000V and the Cisco IOS

The SNMP Version 3 implementation in the ASA 1000V differs from the SNMP Version 3 implementation in the Cisco IOS in the following ways:

- The local-engine and remote-engine IDs are not configurable. The local engine ID is generated when the ASA 1000V starts.
- 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.
- You must create users and groups with the correct security model.
- You must remove users, groups, and hosts in the correct sequence.
- Use of the `snmp-server host` command creates an ASA 1000V rule to allow incoming SNMP traffic.

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.

Failover Guidelines
- Supported in SNMP Version 3.
- The SNMP client in each ASA 1000V 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.

Additional Guidelines
- Does not support view-based access control, but the VACM MIB is available for browsing to determine default view settings.
- 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.

Configuring SNMP

This section describes how to configure SNMP and includes the following topics:
- Enabling SNMP, page 30-9
- Configuring SNMP Traps, page 30-9
- Configuring a CPU Usage Threshold, page 30-10
- Configuring a Physical Interface Threshold, page 30-10
- Using SNMP Version 1 or 2c, page 30-11
- Using SNMP Version 3, page 30-12
Enabling SNMP

The SNMP agent that runs on the ASA 1000V 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:

```
hostname(config)# snmp-server enable
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>snmp-server enable</code></td>
<td>Ensures that the SNMP server on the ASA 1000V is enabled. By default, the SNMP server is enabled.</td>
</tr>
</tbody>
</table>

Example:
```
hostname(config)# snmp-server enable
```

What to Do Next

See the “Configuring SNMP Traps” section on page 30-9.

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:

```
hostname(config)# snmp-server enable traps snmp authentication linkup linkdown coldstart
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>

Example:
```
hostname(config)# snmp-server enable traps snmp authentication linkup linkdown coldstart warmstart
```

Notes:
- 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.
- If the CPU usage is greater than the configured threshold value for the configured monitoring period, the `cpu threshold rising` trap is generated.
- Note: SNMP does not monitor voltage sensors.
### 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><code>snmp cpu threshold rising threshold_value monitoring_period</code></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 <code>no</code> form of this command. If the <code>snmp cpu threshold rising</code> 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>

### 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><code>snmp interface threshold threshold_value</code></td>
<td>Configures the threshold value for an SNMP physical interface. To clear the threshold value for an SNMP physical interface, use the <code>no</code> 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.</td>
</tr>
</tbody>
</table>

### What to Do Next

See the “Configuring a Physical Interface Threshold” section on page 30-10.
# 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>Step</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>`snmp-server host interface) hostname</td>
<td>ip_address} [trap</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><code>snmp-server community community-string</code></td>
<td>Sets the community string, which is for use only with SNMP Version 1 or 2c.</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>`snmp-server [contact</td>
<td>location] text`</td>
</tr>
</tbody>
</table>

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`
- `hostname(config)# snmp-server community onceuponatime`
### What to Do Next

See the “Monitoring SNMP” section on page 30-15.

### 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 30-6. The <code>auth</code> keyword enables packet authentication. The <code>noauth</code> keyword indicates no packet authentication or encryption is being used. The <code>priv</code> keyword enables packet encryption and authentication. No default values exist for the <code>auth</code> or <code>priv</code> keywords.</td>
</tr>
</tbody>
</table>
| `snmp-server group group-name v3 [auth | noauth | priv]` | **Example:**
| `hostname(config)# snmp-server group testgroup1 v3 auth` | **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 ASA 1000V console, enter the following commands:

```
hostname(config)# logging list snmp message 212001-212015
hostname(config)# logging console snmp
```

To make sure 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
```
Troubleshooting Tips

The output is based on the SNMP group of the SNMPv2-MIB.

To make sure that SNMP packets are going through the ASA 1000V and to the SNMP process, enter the following commands:

```plaintext
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 ASA 1000V correctly, use a packet capture to isolate the problem by entering the following commands:

```plaintext
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 ASA 1000V 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 the Cisco TAC.

If SNMP traffic is not being allowed through the ASA 1000V Ethernet interfaces, you might also need to permit ICMP traffic from the remote SNMP server using the `icmp permit` 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 30-6 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>
<thead>
<tr>
<th>Example 1</th>
<th>Example 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>hostname# <code>show interface GigabitEthernet3/2</code></td>
<td></td>
</tr>
<tr>
<td>interface GigabitEthernet3/2</td>
<td></td>
</tr>
<tr>
<td>description fullt-mgmt</td>
<td></td>
</tr>
<tr>
<td>nameif mgmt</td>
<td></td>
</tr>
<tr>
<td>security-level 10</td>
<td></td>
</tr>
<tr>
<td>ip address 10.7.14.201 255.255.255.0</td>
<td></td>
</tr>
<tr>
<td>management-only</td>
<td></td>
</tr>
<tr>
<td>hostname# <code>show traffic</code></td>
<td></td>
</tr>
<tr>
<td>(Condensed output)</td>
<td></td>
</tr>
<tr>
<td>Physical Statistics</td>
<td></td>
</tr>
<tr>
<td>GigabitEthernet3/2:</td>
<td></td>
</tr>
<tr>
<td>received (in 121.760 secs)</td>
<td></td>
</tr>
<tr>
<td>36 packets 3428 bytes</td>
<td></td>
</tr>
<tr>
<td>0 pkts/sec 28 bytes/sec</td>
<td></td>
</tr>
<tr>
<td>Logical Statistics</td>
<td></td>
</tr>
<tr>
<td>mgmt:</td>
<td></td>
</tr>
<tr>
<td>received (in 117.780 secs)</td>
<td></td>
</tr>
<tr>
<td>36 packets 2780 bytes</td>
<td></td>
</tr>
<tr>
<td>0 pkts/sec 23 bytes/sec</td>
<td></td>
</tr>
</tbody>
</table>

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.

The ifIndex value of the management interface:

```
IF_MIB::ifDescr.6 = ASA 1000V 'mgmt' interface
```

The ifInOctets value 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 ASA 1000V. 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 30-16
- **SNMP Monitoring**, page 30-16
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 ASA 1000V to a specified host on a specified interface.

For detailed information about syslog messages, see the 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 or more of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show running-config {default} 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 30-17
- Configuration Example for SNMP Version 3, page 30-17

Configuration Example for SNMP Versions 1 and 2c

The following example shows how the ASA 1000V 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 ohwhatakeyisthee
```

Configuration Example for SNMP Version 3

The following example shows how the ASA 1000V 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 29, “Configuring Logging.”

Additional References

For additional information related to implementing SNMP, see the following sections:

- RFCs for SNMP Version 3, page 30-18
- MIBs, page 30-18
- Application Services and Third-Party Tools, page 30-20

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 ASA 1000V by release, see the following URL:

Not all OIDs in MIBs are supported. To obtain a list of the supported SNMP MIBs and OIDs for a specific ASA 1000V, 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.

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.        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
[6] 1.3.6.1.2.1.1.7.        sysServices
```
<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>[7]</td>
<td>1.3.6.1.2.1.2.1.</td>
<td>ifNumber</td>
</tr>
<tr>
<td>[8]</td>
<td>1.3.6.1.2.1.2.1.1.</td>
<td>ifIndex</td>
</tr>
<tr>
<td>[9]</td>
<td>1.3.6.1.2.1.2.1.2.</td>
<td>ifDescr</td>
</tr>
<tr>
<td>[10]</td>
<td>1.3.6.1.2.1.2.1.3.</td>
<td>ifType</td>
</tr>
<tr>
<td>[11]</td>
<td>1.3.6.1.2.1.2.1.4.</td>
<td>ifMtu</td>
</tr>
<tr>
<td>[12]</td>
<td>1.3.6.1.2.1.2.1.5.</td>
<td>ifSpeed</td>
</tr>
<tr>
<td>[13]</td>
<td>1.3.6.1.2.1.2.1.6.</td>
<td>ifPhysAddress</td>
</tr>
<tr>
<td>[14]</td>
<td>1.3.6.1.2.1.2.1.7.</td>
<td>ifAdminStatus</td>
</tr>
<tr>
<td>[15]</td>
<td>1.3.6.1.2.1.2.1.8.</td>
<td>ifOperStatus</td>
</tr>
<tr>
<td>[16]</td>
<td>1.3.6.1.2.1.2.1.9.</td>
<td>ifLastChange</td>
</tr>
<tr>
<td>[17]</td>
<td>1.3.6.1.2.1.2.1.10.</td>
<td>ifInOctets</td>
</tr>
<tr>
<td>[18]</td>
<td>1.3.6.1.2.1.2.1.11.</td>
<td>ifInUcastPkts</td>
</tr>
<tr>
<td>[19]</td>
<td>1.3.6.1.2.1.2.1.12.</td>
<td>ifInNUcastPkts</td>
</tr>
<tr>
<td>[20]</td>
<td>1.3.6.1.2.1.2.1.13.</td>
<td>ifInDiscards</td>
</tr>
<tr>
<td>[21]</td>
<td>1.3.6.1.2.1.2.2.1.14.</td>
<td>ifInErrors</td>
</tr>
<tr>
<td>[22]</td>
<td>1.3.6.1.2.1.2.2.1.16.</td>
<td>ifOutOctets</td>
</tr>
<tr>
<td>[23]</td>
<td>1.3.6.1.2.1.2.2.1.17.</td>
<td>ifOutUcastPkts</td>
</tr>
<tr>
<td>[24]</td>
<td>1.3.6.1.2.1.2.2.1.18.</td>
<td>ifOutNUcastPkts</td>
</tr>
<tr>
<td>[25]</td>
<td>1.3.6.1.2.1.2.2.1.19.</td>
<td>ifOutDiscards</td>
</tr>
<tr>
<td>[26]</td>
<td>1.3.6.1.2.1.2.2.1.20.</td>
<td>ifOutErrors</td>
</tr>
<tr>
<td>[27]</td>
<td>1.3.6.1.2.1.2.2.1.21.</td>
<td>ifOutQLen</td>
</tr>
<tr>
<td>[28]</td>
<td>1.3.6.1.2.1.2.2.1.22.</td>
<td>ifSpecific</td>
</tr>
<tr>
<td>[29]</td>
<td>1.3.6.1.2.1.4.1.</td>
<td>ipForwarding</td>
</tr>
<tr>
<td>[30]</td>
<td>1.3.6.1.2.1.4.20.1.1.</td>
<td>ipAdEntAddr</td>
</tr>
<tr>
<td>[31]</td>
<td>1.3.6.1.2.1.4.20.1.2.</td>
<td>ipAdEntIfIndex</td>
</tr>
<tr>
<td>[32]</td>
<td>1.3.6.1.2.1.4.20.1.3.</td>
<td>ipAdEntNetMask</td>
</tr>
<tr>
<td>[33]</td>
<td>1.3.6.1.2.1.4.20.1.4.</td>
<td>ipAdEntBcastAddr</td>
</tr>
<tr>
<td>[34]</td>
<td>1.3.6.1.2.1.4.20.1.5.</td>
<td>ipAdEntReasmMaxSize</td>
</tr>
<tr>
<td>[35]</td>
<td>1.3.6.1.2.1.11.1.</td>
<td>snmpInPkts</td>
</tr>
<tr>
<td>[36]</td>
<td>1.3.6.1.2.1.11.2.</td>
<td>snmpOutPkts</td>
</tr>
<tr>
<td>[37]</td>
<td>1.3.6.1.2.1.11.3.</td>
<td>snmpInBadVersions</td>
</tr>
<tr>
<td>[38]</td>
<td>1.3.6.1.2.1.11.4.</td>
<td>snmpInBadCommunityNames</td>
</tr>
<tr>
<td>[39]</td>
<td>1.3.6.1.2.1.11.5.</td>
<td>snmpInBadCommunityUses</td>
</tr>
<tr>
<td>[40]</td>
<td>1.3.6.1.2.1.11.6.</td>
<td>snmpInASNParseErrs</td>
</tr>
<tr>
<td>[41]</td>
<td>1.3.6.1.2.1.11.7.</td>
<td>snmpInTooBigs</td>
</tr>
<tr>
<td>[42]</td>
<td>1.3.6.1.2.1.11.8.</td>
<td>snmpInNoSuchNames</td>
</tr>
<tr>
<td>[43]</td>
<td>1.3.6.1.2.1.11.9.</td>
<td>snmpInBadValues</td>
</tr>
<tr>
<td>[44]</td>
<td>1.3.6.1.2.1.11.10.</td>
<td>snmpInReadOnlys</td>
</tr>
<tr>
<td>[45]</td>
<td>1.3.6.1.2.1.11.11.</td>
<td>snmpInGenErrs</td>
</tr>
<tr>
<td>[46]</td>
<td>1.3.6.1.2.1.11.12.</td>
<td>snmpInTotalReqVars</td>
</tr>
<tr>
<td>[47]</td>
<td>1.3.6.1.2.1.11.13.</td>
<td>snmpInTotalSetVars</td>
</tr>
<tr>
<td>[48]</td>
<td>1.3.6.1.2.1.11.14.</td>
<td>snmpInGetRequests</td>
</tr>
<tr>
<td>[49]</td>
<td>1.3.6.1.2.1.11.15.</td>
<td>snmpInGetNexts</td>
</tr>
<tr>
<td>[50]</td>
<td>1.3.6.1.2.1.11.16.</td>
<td>snmpInSetRequests</td>
</tr>
<tr>
<td>[51]</td>
<td>1.3.6.1.2.1.11.17.</td>
<td>snmpInGetResponses</td>
</tr>
<tr>
<td>[52]</td>
<td>1.3.6.1.2.1.11.18.</td>
<td>snmpInTraps</td>
</tr>
<tr>
<td>[53]</td>
<td>1.3.6.1.2.1.11.19.</td>
<td>snmpOutBigs</td>
</tr>
<tr>
<td>[54]</td>
<td>1.3.6.1.2.1.11.20.</td>
<td>snmpOutNoSuchNames</td>
</tr>
<tr>
<td>[55]</td>
<td>1.3.6.1.2.1.11.21.</td>
<td>snmpOutBadValues</td>
</tr>
<tr>
<td>[56]</td>
<td>1.3.6.1.2.1.11.22.</td>
<td>snmpOutGenErrs</td>
</tr>
<tr>
<td>[57]</td>
<td>1.3.6.1.2.1.11.23.</td>
<td>snmpOutGetRequests</td>
</tr>
<tr>
<td>[58]</td>
<td>1.3.6.1.2.1.11.24.</td>
<td>snmpOutGetNexts</td>
</tr>
<tr>
<td>[59]</td>
<td>1.3.6.1.2.1.11.25.</td>
<td>snmpOutSetRequests</td>
</tr>
<tr>
<td>[60]</td>
<td>1.3.6.1.2.1.11.26.</td>
<td>snmpOutGetResponses</td>
</tr>
<tr>
<td>[61]</td>
<td>1.3.6.1.2.1.11.27.</td>
<td>snmpOutTraps</td>
</tr>
<tr>
<td>[62]</td>
<td>1.3.6.1.2.1.11.28.</td>
<td>snmpEnableAuthenTraps</td>
</tr>
<tr>
<td>[63]</td>
<td>1.3.6.1.2.1.11.29.</td>
<td>snmpSilentDrops</td>
</tr>
<tr>
<td>[64]</td>
<td>1.3.6.1.2.1.11.30.</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>ifInMulticastPkts</td>
</tr>
<tr>
<td>[67]</td>
<td>1.3.6.1.2.1.31.1.1.1.3.</td>
<td>ifInBroadcastPkts</td>
</tr>
<tr>
<td>[68]</td>
<td>1.3.6.1.2.1.31.1.1.1.4.</td>
<td>ifOutMulticastPkts</td>
</tr>
<tr>
<td>[69]</td>
<td>1.3.6.1.2.1.31.1.1.1.5.</td>
<td>ifOutBroadcastPkts</td>
</tr>
<tr>
<td>[70]</td>
<td>1.3.6.1.2.1.31.1.1.1.6.</td>
<td>ifHCInOctets</td>
</tr>
</tbody>
</table>
Application Services and Third-Party Tools

For information about SNMP support, see the following URL:

For information about using third-party tools to walk SNMP Version 3 MIBs, see the following URL:

Feature History for SNMP

Table 30-7 lists each feature change.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>SNMP</td>
<td>8.7(1)</td>
<td>Object identifiers and physical vendor type values were added to support the ASA 1000V.</td>
</tr>
</tbody>
</table>
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 31-3.

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 31-1
- Licensing Requirements for Anonymous Reporting and Smart Call Home, page 31-4
- Prerequisites for Smart Call Home and Anonymous Reporting, page 31-5
- Guidelines and Limitations, page 31-5
- Configuring Anonymous Reporting and Smart Call Home, page 31-6
- Monitoring Smart Call Home, page 31-19
- Configuration Example for Smart Call Home, page 31-19
- Feature History for Anonymous Reporting and Smart Call Home, page 31-20

Information About Anonymous Reporting and Smart Call Home

This section includes the following topics:

- Information About Anonymous Reporting, page 31-2
- Information About Smart Call Home, page 31-4
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.
• **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:

Anonymous—Enables Anonymous Reporting.

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 31-6 or the “Configuring Smart Call Home” section on page 31-7.

### 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 ASA 1000V 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 31-3 and see the “Configuring the DNS Server” section on page 5-10.)

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 31-6
- Configuring Smart Call Home, page 31-7

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><strong>Example:</strong> 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><strong>Example:</strong> 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 31-7
- Declaring and Authenticating a CA Trust Point, page 31-8
- Configuring DNS, page 31-8
- Subscribing to Alert Groups, page 31-9
- Testing Call Home Communications, page 31-11
- Optional Configuration Procedures, page 31-13

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>2</td>
<td>call-home</td>
<td>Enters call-home configuration mode.</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>4</td>
<td>profile profile-name</td>
<td>Enables the profile. The default profile name is CiscoTAC-1.</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>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 31-12.</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</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>crypto ca trustpoint trustpoint-name</td>
<td>Configures a trustpoint and prepares for certificate enrollment.</td>
</tr>
<tr>
<td>3</td>
<td>exit</td>
<td>Exits CA trustpoint configuration mode and returns to global configuration mode.</td>
</tr>
<tr>
<td>4</td>
<td>crypto ca authenticate trustpoint</td>
<td>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.</td>
</tr>
<tr>
<td>5</td>
<td>quit</td>
<td>Specifies the end of the security certificate text and confirms acceptance of entered security certificate.</td>
</tr>
</tbody>
</table>

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:
Chapter 31 Configuring Anonymous Reporting and Smart Call Home

Configuring Anonymous Reporting and Smart Call Home

Subscribing to Alert Groups

An alert group is a predefined subset of the Smart Call Home alerts that are supported on the ASA. Different types of Smart Call Home alerts are grouped into different alert groups depending upon their type.

This section includes the following alert group topics:
- Configuring Periodic Notification, page 31-9
- Information about the Message Severity Threshold, page 31-9
- Configuring Alert Group Subscription, page 31-10

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 31-1). Any message with a value lower than the destination profile’s specified threshold is not sent to the destination.
Table 31-1  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.</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

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> call-home</td>
<td>Enters call-home configuration mode.</td>
</tr>
<tr>
<td>Example: hostname(config) # call-home</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> alert-group (all</td>
<td>configuration</td>
</tr>
<tr>
<td>Example: ciscoasa(cfg-call-home)# alert-group all</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> profile profile-name</td>
<td>Enters the profile configuration submode for the specified destination profile.</td>
</tr>
<tr>
<td>Example: hostname(cfg-call-home)# profile profile1</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> subscribe-to-alert-group configuration (periodic</td>
<td>(daily hh:mm</td>
</tr>
<tr>
<td>Example: hostname(cfg-call-home-profile)# subscribe-to-alert-group configuration periodic weekly Wednesday 23:30</td>
<td>To subscribe to all available alert groups, use the subscribe-to-alert-group all command.</td>
</tr>
</tbody>
</table>
### 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 31-12**
- **Sending a Smart Call Home Alert Group Message Manually, page 31-12**
- **Sending the Output of a Command, page 31-12**

---

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>`subscribe-to-alert-group environment [severity {catastrophic</td>
<td>disaster</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong>&lt;br&gt;hostname(cfg-call-home-profile)# subscribe-to-alert-group examplealertgroupname severity critical</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>`subscribe-to-alert-group syslog [severity {catastrophic</td>
<td>disaster</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong>&lt;br&gt;hostname(cfg-call-home-profile)# subscribe-to-alert-group syslog severity notification pattern UPDOWN</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>`subscribe-to-alert-group inventory [periodic {daily hh:mm</td>
<td>monthly date hh:mm</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong>&lt;br&gt;hostname(cfg-call-home-profile)# subscribe-to-alert-group inventory periodic daily 06:30</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>`subscribe-to-alert-group telemetry periodic {hourly</td>
<td>daily</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong>&lt;br&gt;hostname(cfg-call-home-profile)# subscribe-to-alert-group monthly 15</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>`subscribe-to-alert-group snapshot periodic {interval minutes</td>
<td>hourly</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong>&lt;br&gt;hostname(cfg-call-home-profile)# subscribe-to-alert-group snapshot periodic interval weekly wednesday 23:15</td>
<td></td>
</tr>
</tbody>
</table>
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>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`call-home send alert-group {inventory</td>
<td>configuration</td>
<td>snapshot</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 31-13
- Configuring the Mail Server, page 31-15
- Configuring Call Home Traffic Rate Limiting, page 31-15
- Destination Profile Management, page 31-16

### 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</td>
<td>call-home</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname(config)# call-home</td>
</tr>
<tr>
<td>Step 2</td>
<td>contact-email-addr email-address</td>
</tr>
<tr>
<td>Example:</td>
<td>ciscoasa(cfg-call-home)# contact-email-addr <a href="mailto:username@example.com">username@example.com</a></td>
</tr>
<tr>
<td>Step 3</td>
<td>(Optional) phone-number phone-number-string</td>
</tr>
<tr>
<td>Example:</td>
<td>ciscoasa(cfg-call-home)# phone-number 8005551122</td>
</tr>
<tr>
<td>Step 4</td>
<td>(Optional) street-address street-address</td>
</tr>
<tr>
<td>Example:</td>
<td>ciscoasa(cfg-call-home)# street-address “1234 Any Street, Any city, Any state, 12345”</td>
</tr>
</tbody>
</table>
Configuring Anonymous Reporting and Smart Call Home

Step 5 *(Optional)*  
**contact-name contact name**

*Example:*
```
ciscoasa(cfg-call-home)# contact-name contactname1234
```

Step 6 *(Optional)*  
**customer-id customer-id-string**

*Example:*
```
ciscoasa(cfg-call-home)# customer-id customer1234
```

Step 7 *(Optional)*  
**site-id site-id-string**

*Example:*
```
ciscoasa(cfg-call-home)# site-id site1234
```

Step 8 *(Optional)*  
**contract-id contract-id-string**

*Example:*
```
ciscoasa(cfg-call-home)# contract-id contract1234
```

This example shows the configuration of contact information:

```
hostname# configure terminal
hostname(config)# call-home
ciscoasa(cfg-call-home)# contact-email-addr username@example.com
phone-number 8005551122
street-address "1234 Any Street, Any city, Any state, 12345"
ciscoasa(cfg-call-home)# contact-name contactname1234
ciscoasa(cfg-call-home)# customer-id customer1234
ciscoasa(cfg-call-home)# site-id site1234
ciscoasa(cfg-call-home)# contract-id contract1234
```
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>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>call-home</td>
</tr>
<tr>
<td>Example:</td>
<td>Enters call home configuration mode.</td>
</tr>
<tr>
<td>hostname(config)# call-home</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 2</th>
<th>mail-server ip-address</th>
<th>name priority 1-100 all</th>
</tr>
</thead>
<tbody>
<tr>
<td>Example:</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>
<td></td>
</tr>
<tr>
<td>ciscoasa(cfg-call-home)# mail-server 10.10.1.1 smtp.example.com priority 1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

This example shows the configuration of a primary mail server (named “smtp.example.com”) and a secondary mail server at IP address 10.10.1.1:

hostname# configure terminal
hostname(config)# call-home
ciscoasa(cfg-call-home)# mail-server smtp.example.com priority 1
ciscoasa(cfg-call-home)# mail-server 10.10.1.1 priority 2
ciscoasa(cfg-call-home)# exit
hostname(config)#

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>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>call-home</td>
</tr>
<tr>
<td>Example:</td>
<td>Enters call home configuration mode.</td>
</tr>
<tr>
<td>hostname(config)# call-home</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 2</th>
<th>rate-limit msg-count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Example:</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>
<tr>
<td>ciscoasa(cfg-call-home)# rate-limit 5</td>
<td></td>
</tr>
</tbody>
</table>
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 31-16
- Activating and Deactivating a Destination Profile, page 31-17
- Copying a Destination Profile, page 31-18
- Renaming a Destination Profile, page 31-18

#### Configuring a Destination Profile

To configure a destination profile for e-mail or for HTTP, perform this task:

<table>
<thead>
<tr>
<th>Step 1</th>
<th><strong>call-home</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Example:</td>
<td>hostname(config)# call-home</td>
</tr>
<tr>
<td>Enters call home configuration mode.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 2</th>
<th><strong>profile profile-name</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Example:</td>
<td>hostname(cfg-call-home)# profile newprofile</td>
</tr>
<tr>
<td>Enters the profile configuration mode for the specified destination profile. If the specified destination profile does not exist, it is created.</td>
<td></td>
</tr>
</tbody>
</table>

| Step 3 | **destination (email address | http url) | message-size-limit size | preferred-msg-format (long-text | short-text | xml) transport-method (email | http))** |
|--------|-----------------------------|
| Example: | hostname(cfg-call-home-profile)# destination address email username@example.com |
| hostname (cfg-call-home-profile)# destination preferred-msg-format long-text |
| Configures the destination, message size, message format, and transport method for the smart call-home message receiver. The default message format is XML, and the default enabled transport method is e-mail. The e-mail-address is the e-mail address of the smart call-home receiver, which can be up to 100 characters long. By default, the maximum URL size is 5 MB. Use the short-text format to send and read a message on a mobile device, and use the long text format to send and read a message on a computer. If the message receiver is the Smart Call Home back-end server, ensure that the preferred-msg-format is XML, as the back-end server can accept messages in XML format only. |
### 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>Example:</td>
<td>hostname(config)# call-home</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>Example:</td>
<td>hostname(cfg-call-home)# profile newprofile</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>Example:</td>
<td>ciscoasa(cfg-call-home-profile)# active</td>
</tr>
<tr>
<td>4</td>
<td>no active</td>
<td>Disables the destination profile.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td>ciscoasa(cfg-call-home-profile)# no active</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
```
## Configuring Anonymous Reporting and Smart Call Home

### Copying a Destination Profile
To create a new destination profile by copying an existing profile, perform this task:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> 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><strong>Step 2</strong> profile profilename</td>
<td>Specifies the profile to copy.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>ciscoasa(cfg-call-home)# profile newprofile</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> 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>Example:</td>
<td></td>
</tr>
<tr>
<td>ciscoasa(cfg-call-home)# copy profile profile1 profile2</td>
<td></td>
</tr>
</tbody>
</table>

This example shows how to copy an existing profile:

```plaintext
hostname# configure terminal
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>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> 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><strong>Step 2</strong> profile profilename</td>
<td>Specifies the profile to rename.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>ciscoasa(cfg-call-home)# profile newprofile</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> 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>Example:</td>
<td></td>
</tr>
<tr>
<td>ciscoasa(cfg-call-home)# rename profile profile1 profile2</td>
<td></td>
</tr>
</tbody>
</table>

This example shows how to rename an existing profile:

```plaintext
hostname# configure terminal
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 ( \text{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 31-2 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 31-2  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 ASA 1000V, 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>
PART 11

System Administration
Managing Software and Configurations

This chapter describes how to manage the ASA 1000V software and configurations and includes the following sections:

- Managing the Flash File System, page 32-1
- Downloading Software or Configuration Files to Flash Memory, page 32-2
- Configuring the Application Image and ASDM Image to Boot, page 32-4
- Configuring the File to Boot as the Startup Configuration, page 32-4
- Performing Zero Downtime Upgrades for Failover Pairs, page 32-5
- Backing Up Configuration Files or Other Files, page 32-6
- Configuring Auto Update Support, page 32-7

Managing the Flash File System

This section includes the following topics:

- Viewing Files in Flash Memory, page 32-1
- Deleting Files from Flash Memory, page 32-2

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:

  Enter disk0: for the internal flash memory.

  For example:

  hostname# dir

  Directory of disk0:/
  500   -rw-  4958208     22:56:20 Nov 29 2004  cdisk.bin
  2513  -rw-  4634        19:32:48 Sep 17 2004  first-backup
  2788  -rw-  21601       20:51:46 Nov 23 2004  backup.cfg
  2927  -rw-  8670632     20:42:48 Dec 08 2004  asdmfile.bin

- To view extended information about a specific file, enter the following command:
hostname# show file information [path:]filename

The default path is the root directory of the internal flash memory (disk0:/).

For example:
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:
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 ASA 1000V, 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, the following error message appears:

%Error opening disk0:/Config.cfg (File exists).

This section includes the following topics:
- Downloading a File to a Specific Location, page 32-2
- Downloading a File to the Startup or Running Configuration, page 32-3

Downloading a File to a Specific Location

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 32-3.

To configure the ASA 1000V 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 32-4.
To configure the ASA 1000V to use a specific configuration as the startup configuration, see the “Configuring the File to Boot as the Startup Configuration” section on page 32-4.

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:/{path/}filename
  ```

- To copy from an FTP server, enter the following command:
  
  ```
  hostname# copy ftp://[user[:password]@]server[/path]/filename disk0:/{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:/{path/}filename
  ```

- To copy from an SMB server, enter the following command:
  
  ```
  hostname# copy smb://[user[:password]@]server[/path]/filename disk0:/{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.

## Downloading a File to the Startup or Running Configuration

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:

- 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:
Configuring the Application Image and ASDM Image to Boot

By default, the ASA 1000V 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 ASA 1000V 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:[path/][filename]`
- `tftp://[user[:password]@[server[/port/]]][path/][filename]`

You can enter up to four `boot system` command entries to specify different images to boot from in order; the ASA 1000V boots the first image it finds. Only one `boot system tftp` command can be configured, and it must be the first one configured.

To configure the ASDM image to boot, enter the following command:

```
hostname(config)# asdm image disk0:[path/][filename]
```

Configuring the File to Boot as the Startup Configuration

By default, the ASA 1000V 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:[path/][filename]
```
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 32-1 shows the supported scenarios for performing zero-downtime upgrades on a failover pair.

Table 32-1  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 8.4(1) to 8.4(3) 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 8.0(1) to 8.1(1). Upgrading from 8.0(1) directly to 8.2(1) is not supported for zero-downtime upgrades; you must first upgrade to 8.1(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 8.3.x to 8.4.x.

For more details about upgrading the software on a failover pair, see the following topics:

- Upgrading an Active/Standby Failover Configuration, page 32-5
- Backing Up Configuration Files or Other Files, page 32-6

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 32-4).

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.
Note Use the `show failover` command to verify that the standby unit is in the Standby Ready state.

```
active# no failover active
```

**Step 4** Reload the former active unit (now the new standby unit) by entering the following command:

```
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:

```
newstandby# failover active
```

---

## Backing Up Configuration Files or Other Files

This section includes the following topics:

- Backing up the Configuration, page 32-6
- Copying the Configuration from the Terminal Display, page 32-6

### Backing up the Configuration

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} disk0:[/path/]filename
  ```

**Note** Be sure that the destination directory exists. If it does not exist, first create the directory using the `mkdir` command.

### 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.
Configuring Auto Update Support

Auto Update is a protocol specification that allows an Auto Update Server to download configurations and software images to many ASA 1000Vs and can provide basic monitoring of the ASA 1000Vs from a central location.

The ASA 1000V 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 ASA 1000Vs configured as Auto Update clients.

This section includes the following topics:

- Configuring Communication with an Auto Update Server, page 32-7
- Configuring Client Updates as an Auto Update Server, page 32-9
- Viewing Auto Update Status, page 32-9

Configuring Communication with an Auto Update Server

To configure the ASA 1000V 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]
```

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 site-to-site VPN tunnel used for management access.

**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 ASA 1000V serial number.
- The `hostname` argument specifies the ASA 1000V 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 ASA 1000V 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 ASA 1000V has the most recent image and configuration. This condition is reported with syslog message 201008.

In the following example, an ASA 1000V is configured to poll an Auto Update Server with the IP address 209.165.200.224, at port number 1742, from the outside interface.

The ASA 1000V 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 ASA 1000V 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`

`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 ASA 1000Vs configured as Auto Update clients and lets you specify the type of software component (ASDM or boot image), the type or family of ASA 1000V, revision numbers to which the update applies, and a URL or IP address from which to obtain the update.

To configure the ASA 1000V 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 ASA 1000Vs:

```
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 ASA 1000V.
- 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 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 `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 ASA 1000Vs of a particular type. That is, specify the type of ASA 1000V 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 ASA 1000V 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 ASA 1000Vs, enter the following command:

```
hostname(config)# client-update type asa1000V component asdm url http://192.168.1.114/aus/asdm671.bin rev-nums 8.7(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
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 2011
Last PDM update: 23:36:46 PST Tue Nov 12 2011
Troubleshooting

This chapter describes how to troubleshoot the ASA 1000V and includes the following sections:

- Testing Your Configuration, page 33-1
- Reloading the ASA 1000V, page 33-8
- Performing Password Recovery, page 33-8
- Erasing the Flash File System, page 33-9
- Other Troubleshooting Tools, page 33-10

Testing Your Configuration

This section describes how to test connectivity, how to ping the ASA 1000V Ethernet 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 ASA 1000V, follow the steps in the “Disabling the Test Configuration” section on page 33-7.

This section includes the following topics:

- Enabling ICMP Debugging Messages and Syslog Messages, page 33-2
- Pinging ASA 1000V Interfaces, page 33-3
- Passing Traffic Through the ASA 1000V, page 33-5
- Disabling the Test Configuration, page 33-7
- Determining Packet Routing with Traceroute, page 33-7
- Tracing Packets with Packet Tracer, page 33-7
- Handling TCP Packet Loss, page 33-8
Enabling ICMP Debugging Messages and Syslog Messages

Debugging messages and syslog messages can help you troubleshoot why your pings are not successful. The ASA 1000V only shows ICMP debugging messages for pings to the ASA 1000V Ethernet interfaces, and not for pings through the ASA 1000V 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><code>debug icmp trace</code></td>
<td>Shows ICMP packet information for pings to the ASA 1000V Ethernet 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><code>logging monitor debug</code></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></td>
</tr>
<tr>
<td>You can alternately use the <code>logging buffer debug</code> command to send log messages to a buffer, and then view them later using the <code>show logging</code> command.</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td></td>
</tr>
<tr>
<td><code>terminal monitor</code></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><code>logging on</code></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><code>policy-map name</code></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><code>class classmap_name</code></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><code>inspect icmp</code></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 ASA 1000V 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 ASA 1000V Interfaces

To test whether the ASA 1000V interfaces are up and running and that the ASA 1000V and connected routers are operating correctly, you can ping the ASA 1000V interfaces. To ping the ASA 1000V interfaces, perform the following steps:

<table>
<thead>
<tr>
<th>Step 1</th>
<th>Draw a diagram of your ASA 1000V that shows the interface names, security levels, and IP addresses.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Note</td>
<td>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.</td>
</tr>
</tbody>
</table>

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 ASA 1000V. You will use this information in this procedure and in the procedure in the “Passing Traffic Through the ASA 1000V” section on page 33-5. (See Figure 33-1.)
Step 2  Ping each ASA 1000V interface from the directly connected routers. This test ensures that the ASA 1000V interfaces are active and that the interface configuration is correct.

A ping might fail if the ASA 1000V interface is not active, the interface configuration is incorrect, or if a switch between the ASA 1000V and a router is down (see Figure 33-2). In this case, no debugging messages or syslog messages appear, because the packet never reaches the ASA 1000V.

If the ping reaches the ASA 1000V, 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 33-3).
Step 3 Ping each ASA 1000V interface from a remote host. This test checks whether the directly connected router can route the packet between the host and the ASA 1000V, and whether the ASA 1000V can correctly route the packet back to the host.

A ping might fail if the ASA 1000V does not have a return route to the host through the intermediate router (see Figure 33-4). 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 ASA 1000V

After you successfully ping the ASA 1000V interfaces, make sure that traffic can pass successfully through the ASA 1000V. This test shows that NAT is operating correctly, if configured.

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>Step 1</td>
<td>access-list ICMPACL extended permit icmp any any</td>
</tr>
<tr>
<td></td>
<td>Example: hostname(config)# access-list ICMPACL extended permit icmp any any</td>
</tr>
<tr>
<td></td>
<td>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.</td>
</tr>
<tr>
<td>Step 2</td>
<td>access-group ICMPACL in interface interface_name</td>
</tr>
<tr>
<td></td>
<td>Example: hostname(config)# access-group ICMPACL in interface inside</td>
</tr>
</tbody>
</table>
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 ASA 1000V.

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 (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.

The ping might fail because NAT is not configured correctly (see Figure 33-5). 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 ASA 1000V only shows ICMP debugging messages for pings to the ASA 1000V Ethernet interfaces, and not for pings through the ASA 1000V to other hosts.

![Figure 33-5 Ping Failure Because the ASA 1000V 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 ASA 1000V and that prints debugging messages. If you leave this configuration in place, it can pose a serious security risk. Debugging messages also slow the ASA 1000V performance.

To disable the test configuration, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>no debug icmp trace</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname (config)# no debug icmp trace</td>
</tr>
<tr>
<td>Step 2</td>
<td>no logging on</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname (config)# no logging on</td>
</tr>
<tr>
<td>Step 3</td>
<td>no access-list ICMPACL</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname (config)# no access-list ICMPACL</td>
</tr>
<tr>
<td>Step 4</td>
<td>no service-policy ICMP-POLICY</td>
</tr>
<tr>
<td>Example:</td>
<td>hostname (config)# no service-policy ICMP-POLICY</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 ASA 1000V.

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 ASA 1000V. 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 ASA 1000V 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 IP address based on the user identity and the FQDN.

To trace packets, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>packet-tracer input [ifc_name] [icmp [sip</td>
<td>user username</td>
</tr>
<tr>
<td>tcp</td>
<td>[sip</td>
</tr>
<tr>
<td>udp</td>
<td>[sip</td>
</tr>
<tr>
<td>[rawip [sip</td>
<td>user username</td>
</tr>
<tr>
<td>[detailed]</td>
<td></td>
</tr>
<tr>
<td>[xml]</td>
<td></td>
</tr>
</tbody>
</table>

Example:
hostname# packet-tracer input inside tcp 10.2.25.3 www 209.165.202.158 aol detailed

### Handling TCP Packet Loss

To troubleshoot TCP packet loss, see the “Customizing the TCP Normalizer with a TCP Map” section on page 24-6 for more information.

### Reloading the ASA 1000V

To reload the ASA 1000V, enter the following command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>reload</td>
<td>Restarts the ASA 1000V.</td>
</tr>
</tbody>
</table>

Example:
hostname (config)# reload

### Performing Password Recovery

This section includes the following topics:

• Recovering Passwords or Images on the ASA 1000V, page 33-9
• Disabling Password Recovery, page 33-9
## Recovering Passwords or Images on the ASA 1000V

To recover passwords or images on the ASA 1000V, perform the following steps:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1**  
  `copy running-config filename`  
  Example:  
  `hostname# copy running-config backup.cfg` | Copies the running configuration to a backup file on the ASA 1000V.       |
| **Step 2**  
  `reload`  
  Example:  
  `hostname# reload` | Restarts the ASA 1000V.                                                 |
| **Step 3**  
  GNU GRUB version 2.0(12)4  
  `bootflash:/asa100123-20-smp-k8.bin`  
  `bootflash: /asa100123-20-smp-k8.bin with no configuration load`  
  Example:  
  GNU GRUB version 2.0(12)4  
  `bootflash: /asa100123-20-smp-k8.bin with no configuration load` | From the GNU GRUB menu, press the down arrow, choose the `<filename> with no configuration load` option, then press Enter. The filename is the default boot image filename on the ASA 1000V. The default boot image is never automatically booted through the `fallback` command.  
  Boots the selected boot image. |
| **Step 4**  
  `copy filename running-config`  
  Example:  
  `hostname (config)# copy backup.cfg running-config` | Copies the backup configuration file to the running configuration.         |
| **Step 5**  
  `enable password`  
  Example:  
  `hostname (config)# enable password cisco123` | Resets the password.                                                     |
| **Step 6**  
  `write mem`  
  Example:  
  `hostname (config)# write mem` | Saves the new configuration.                                             |

### Disabling Password Recovery

You cannot disable password recovery on the ASA 1000V.

### Erasing the Flash File System

To erase the flash file system, perform the following steps:

**Step 1**  
Connect to the ASA 1000V console port according to the instructions in “Accessing the ASA 1000V Command-Line Interface” section on page 2-2.
Step 2 Enter the **format** command in privileged EXEC mode as follows:

```
hostname# format disk0: disk1:
```

### Other Troubleshooting Tools

The ASA 1000V provides other troubleshooting tools that you can use. This section includes the following topics:

- Viewing Debugging Messages, page 33-10
- Capturing Packets, page 33-11
- Viewing the Crash Dump, page 33-11
- Coredump, page 33-11
- Monitoring Per-Process CPU Usage, page 33-11

### 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
of less network traffic and fewer users. Debugging during these periods decreases the likelihood that increased `debug` command processing overhead will affect system use. To enable debugging messages, see the `debug` commands in the command reference.

**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 ASA 1000V 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 ASA 1000V. 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.
Using the Command-Line Interface

This appendix describes how to use the CLI on the ASA 1000V 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-7

Note

The CLI uses similar syntax and other conventions to the Cisco IOS CLI, but the ASA 1000V 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 ASA 1000V.

Firewall Mode and Security Context Mode

The ASA 1000V runs in the following modes:

- Routed firewall mode
  The firewall mode determines if the ASA 1000V runs as a Layer 2 or Layer 3 firewall.
- Single context mode

Some commands are only available in certain modes.
Command Modes and Prompts

The ASA 1000V 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

The prompt changes depending on the access mode:

- User EXEC mode
  User EXEC mode lets you see minimum ASA 1000V settings. The user EXEC mode prompt appears as follows when you first access the ASA 1000V:
  hostname>

- 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#

- Global configuration mode
  Global configuration mode lets you change the ASA 1000V 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)#
Syntax Formatting

Command syntax descriptions use the conventions listed in Table A-1.

Table A-1 Syntax Conventions

<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><strong>italics</strong></td>
<td>Italic text indicates arguments for which you supply values.</td>
</tr>
<tr>
<td>[x]</td>
<td>Square brackets enclose an optional element (keyword or argument).</td>
</tr>
<tr>
<td>l</td>
<td>A vertical bar indicates a choice within an optional or required set of keywords or arguments.</td>
</tr>
<tr>
<td>[x</td>
<td>y]</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 t` to view the configuration instead of entering the full command `write terminal`, or you can enter `en` to start privileged mode and `conf t` to start configuration mode. In addition, you can enter `0` to represent `0.0.0.0`.

Command-Line Editing

The ASA 1000V 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 ASA 1000V 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 ASA 1000V 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 ASA 1000V 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 ASA 1000V 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 `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 **metacharacters** have special meaning when used in regular expressions.

Use `Ctrl+V` to escape all of the special characters in the CLI, such as a question mark (`?`) or a tab. For example, type `d[Ctrl+V]?g` to enter `d?g` in the configuration.

For a list of metacharacters, see Table 8-1 on page 8-11.

### Command Output Paging

For commands such as `help` or `?`, `show`, `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 `pager` command lets you choose the number of lines to display before the More prompt appears.

When paging is enabled, the following prompt appears:

```
<--- More --->
```

The More prompt uses syntax similar to the UNIX `more` command:

- To view another screen, press the `Space` bar.
- To view the next line, press the `Enter` key.
- To return to the command line, press the `q` key.

### Adding Comments

You can precede a line with a colon (`:`) 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 `show history` command or by pressing an arrow key to retrieve a previous command, but because the comment is not in the configuration, the `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 ASA 1000V, and includes the following topics:

- **How Commands Correspond with Lines in the Text File, page A-6**
- **Command-Specific Configuration Mode Commands, page A-6**
- **Automatic Text Entries, page A-7**
- **Line Order, page A-7**
- **Commands Not Included in the Text Configuration, page A-7**
- **Passwords, 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 ASA 1000V, it inserts some lines automatically. For example, the ASA 1000V 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 ASA 1000V in its encrypted form, but you cannot unencrypt the passwords yourself.

If you enter an unencrypted password in a text file, the ASA 1000V does not automatically encrypt it when you copy the configuration to the ASA 1000V. The ASA 1000V only encrypts it when you save the running configuration from the command line using the `copy running-config startup-config` or `write memory` command.

Supported Character Sets

The ASA 1000V 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 ASA 1000V CLI supports up to 255 characters (multibyte characters) of ISO 8859-1.
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
- Protocols and Applications, page B-5
- TCP and UDP Ports, page B-5
- Local Ports and Protocols, page B-8
- ICMP Types, page B-9

IPv4 Addresses and Subnet Masks

This section describes how to use IPv4 addresses in the ASA 1000V. 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.
- Class B addresses (128.0.xxx.xxx through 191.255.xxx.xxx) use the first two octets as the network prefix.
IPv4 Addresses and Subnet Masks

- 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 11111111.11111111.11111111.00000000, 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 B-1 Hosts, Bits, and Dotted-Decimal Masks

<table>
<thead>
<tr>
<th>Hosts</th>
<th>/Bits Mask</th>
<th>Dotted-Decimal Mask</th>
</tr>
</thead>
<tbody>
<tr>
<td>16,777,216</td>
<td>/8</td>
<td>255.0.0.0 Class A Network</td>
</tr>
<tr>
<td>65,536</td>
<td>/16</td>
<td>255.255.0.0 Class B Network</td>
</tr>
<tr>
<td>32,768</td>
<td>/17</td>
<td>255.255.128.0</td>
</tr>
<tr>
<td>16,384</td>
<td>/18</td>
<td>255.255.192.0</td>
</tr>
<tr>
<td>8192</td>
<td>/19</td>
<td>255.255.224.0</td>
</tr>
<tr>
<td>4096</td>
<td>/20</td>
<td>255.255.240.0</td>
</tr>
<tr>
<td>2048</td>
<td>/21</td>
<td>255.255.248.0</td>
</tr>
<tr>
<td>1024</td>
<td>/22</td>
<td>255.255.252.0</td>
</tr>
<tr>
<td>512</td>
<td>/23</td>
<td>255.255.254.0</td>
</tr>
<tr>
<td>256</td>
<td>/24</td>
<td>255.255.255.0 Class C Network</td>
</tr>
<tr>
<td>128</td>
<td>/25</td>
<td>255.255.255.128</td>
</tr>
<tr>
<td>64</td>
<td>/26</td>
<td>255.255.255.192</td>
</tr>
<tr>
<td>32</td>
<td>/27</td>
<td>255.255.255.224</td>
</tr>
<tr>
<td>16</td>
<td>/28</td>
<td>255.255.255.240</td>
</tr>
<tr>
<td>8</td>
<td>/29</td>
<td>255.255.255.248</td>
</tr>
<tr>
<td>4</td>
<td>/30</td>
<td>255.255.255.252</td>
</tr>
<tr>
<td>Do not use</td>
<td>/31</td>
<td>255.255.255.254</td>
</tr>
<tr>
<td>1</td>
<td>/32</td>
<td>255.255.255.255 Single Host Address</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 B-2 Class C-Size Network Address

<table>
<thead>
<tr>
<th>Subnet with Mask /29 (255.255.255.248)</th>
<th>Address Range1</th>
</tr>
</thead>
<tbody>
<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>
IPv4 Addresses and Subnet Masks

Table B-2  Class C-Size Network Address (continued)

<table>
<thead>
<tr>
<th>Subnet with Mask /29 (255.255.255.248)</th>
<th>Address Range¹</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.168.0.16</td>
<td>192.168.0.16 to 192.168.0.31</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>192.168.0.248</td>
<td>192.168.0.248 to 192.168.0.255</td>
</tr>
</tbody>
</table>

1. The first and last address of a subnet are reserved. In the first subnet example, you cannot use 192.168.0.0 or 192.168.0.7.

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 B-3  Subnets of Network

<table>
<thead>
<tr>
<th>Subnet with Mask /20 (255.255.240.0)</th>
<th>Address Range¹</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></td>
<td></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.
Protocols and Applications

Table B-4 lists the protocol literal values and port numbers; either can be entered in ASA 1000V commands.

Table B-4  Protocol Literal Values

<table>
<thead>
<tr>
<th>Literal</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<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>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>pcp</td>
<td>108</td>
<td>Payload Compression Protocol.</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-5 lists the literal values and port numbers; either can be entered in ASA 1000V commands. See the following caveats:

- The ASA 1000V 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 ASA 1000V listens for RADIUS on ports 1645 and 1646. If your RADIUS server uses the standard ports 1812 and 1813, you can configure the ASA 1000V 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 ASA 1000V 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
### Table B-5 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>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>
</tbody>
</table>
### TCP and UDP Ports

<table>
<thead>
<tr>
<th>Literal</th>
<th>TCP or UDP?</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<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>
<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>
</tbody>
</table>
Local Ports and Protocols

Table B-5 lists the protocols, TCP ports, and UDP ports that the ASA 1000V may open to process traffic destined to the ASA 1000V. Unless you enable the features and services listed in Table B-6, the ASA 1000V does not open any local protocols or any TCP or UDP ports. You must configure a feature or service for the ASA 1000V 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-6 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>105</td>
<td>N/A</td>
<td></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>1</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>IGMP</td>
<td>2</td>
<td>N/A</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>50</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>IPsec over UDP (NAT-T)</td>
<td>UDP</td>
<td>4500</td>
<td></td>
</tr>
<tr>
<td>IPsec over UDP</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>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 (non-secure)</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td></td>
<td>9 (secure)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Telnet</td>
<td>TCP</td>
<td>23</td>
<td></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>
ICMP Types

Table B-7 lists the ICMP type numbers and names that you can enter in ASA 1000V 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>
APPENDIX C

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 ASA 1000V. Before you configure the ASA 1000V to use an external server, you must configure the server with the correct ASA 1000V 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
- Configuring an External TACACS+ Server, page C-37

Understanding Policy Enforcement of Permissions and Attributes

The ASA 1000V supports several methods of applying user authorization attributes (also called user entitlements or permissions) to VPN connections. You can configure the ASA 1000V to obtain user attributes from a Dynamic Access Policy (DAP) on the ASA 1000V, from an external authentication and/or authorization AAA server (RADIUS or LDAP), from a group policy on the ASA 1000V, or from all three.

If the ASA 1000V 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 ASA 1000V applies attributes in the following order (see Figure C-1).

1. DAP attributes on the ASA 1000V—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 ASA 1000V (User Accounts in ASDM).
3. Group policy configured on the ASA 1000V—If a RADIUS server returns the value of the RADIUS CLASS attribute IETF-Class-25 (OU=group-policy) for the user, the ASA 1000V 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 ASA 1000V 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 ASA 1000V 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 ASA 1000V (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 ASA 1000V 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 ASA 1000V for LDAP Operations, page C-3
The specific steps of these processes vary, depending on which type of LDAP server that you are using.

**Note**
For more information about the LDAP protocol, see RFCs 1777, 2251, and 2849.

## Organizing the ASA 1000V for LDAP Operations

This section describes how to search within the LDAP hierarchy and perform authenticated binding to the LDAP server on the ASA 1000V and includes the following topics:

- Searching the LDAP Hierarchy, page C-3
- Binding the ASA 1000V 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 ASA 1000V lets you tailor the search within the LDAP hierarchy. You configure the following three fields on the ASA 1000V 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 ASA 1000V.

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 ASA 1000V 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 ASA 1000V 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 ASA 1000V to the LDAP Server

Some LDAP servers (including the Microsoft Active Directory server) require the ASA 1000V to establish a handshake via authenticated binding before they accept requests for any other LDAP operations. The ASA 1000V 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 ASA 1000V 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 ASA 1000V 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 ASA 1000V 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 ASA 1000V 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 ASA 1000V does not support anonymous authentication.

Note
As an LDAP client, the ASA 1000V does not support the transmission of anonymous binds or requests.

Defining the ASA 1000V 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

Note
The ASA 1000V 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 ASA 1000Vs. The table includes attribute support information for the VPN 3000 concentrator and PIX 500 series ASA 1000Vs to assist you in configuring networks with a combination of these devices.
## Table C-2  
ASA 1000V 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: ASA 1000V 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   |     | 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 new line 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 ASA 1000V setting                                                                   |
| IE-Proxy-Server           | Y        | Y   | Y   | Integer     | Single                 | IP address                                                                                     |
| IETF-Radius-Class         | Y        | Y   | Y   | 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 ASA 1000V. 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 ASA 1000V 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>IETF-Radius-Service-Type</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>1 = Login &lt;br&gt; 2 = Framed &lt;br&gt; 5 = Remote access &lt;br&gt; 6 = Administrative &lt;br&gt; 7 = NAS prompt</td>
</tr>
<tr>
<td>IETF-Radius-Session-Timeout</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>Seconds</td>
</tr>
<tr>
<td>IKE-Keep-Alives</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Boolean</td>
<td>Single</td>
<td>0 = Disabled &lt;br&gt; 1 = Enabled</td>
</tr>
<tr>
<td>IPsec-Allow-Passwd-Store</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Boolean</td>
<td>Single</td>
<td>0 = Disabled &lt;br&gt; 1 = Enabled</td>
</tr>
<tr>
<td>IPsec-Authentication</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>0 = None &lt;br&gt; 1 = RADIUS &lt;br&gt; 2 = LDAP (authorization only) &lt;br&gt; 3 = NT Domain &lt;br&gt; 4 = SDI (RSA) &lt;br&gt; 5 = Internal &lt;br&gt; 6 = RADIUS with Expiry &lt;br&gt; 7 = Kerberos or Active Directory</td>
</tr>
<tr>
<td>IPsec-Auth-On-Rekey</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Boolean</td>
<td>Single</td>
<td>0 = Disabled &lt;br&gt; 1 = Enabled</td>
</tr>
<tr>
<td>IPsec-Backup-Server-List</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>String</td>
<td>Single</td>
<td>Server addresses (space delimited)</td>
</tr>
<tr>
<td>IPsec-Backup-Servers</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>String</td>
<td>Single</td>
<td>1 = Use client-configured list &lt;br&gt; 2 = Disabled and clear client list &lt;br&gt; 3 = Use backup server list</td>
</tr>
<tr>
<td>IPsec-Client-Firewall-Filter-Name</td>
<td>Y</td>
<td></td>
<td></td>
<td>String</td>
<td>Single</td>
<td>Specifies the name of the filter to be pushed to the client as firewall policy.</td>
</tr>
<tr>
<td>IPsec-Client-Firewall-Filter-Optional</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Required &lt;br&gt; 1 = Optional</td>
</tr>
<tr>
<td>IPsec-Default-Domain</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>String</td>
<td>Single</td>
<td>Specifies the single default domain name to send to the client (1 - 255 characters).</td>
</tr>
<tr>
<td>IPsec-Extended-AUTH-On-Rekey</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>String</td>
<td>Single</td>
<td>String</td>
</tr>
<tr>
<td>IPsec-IKE-Peer-ID-Check</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>1 = Required &lt;br&gt; 2 = If supported by peer certificate &lt;br&gt; 3 = Do not check</td>
</tr>
<tr>
<td>IPsec-IP-Compression</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled &lt;br&gt; 1 = Enabled</td>
</tr>
<tr>
<td>IPsec-Mode-Config</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Boolean</td>
<td>Single</td>
<td>0 = Disabled &lt;br&gt; 1 = Enabled</td>
</tr>
<tr>
<td>IPsec-Over-UDP</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Boolean</td>
<td>Single</td>
<td>0 = Disabled &lt;br&gt; 1 = Enabled</td>
</tr>
</tbody>
</table>
### Table C-2  ASA 1000V 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>
</tbody>
</table>
| IPsec-Required-Client-Firewall-Capability | Y        | Y   | Y   | Integer     | Single                 | 0 = None  
1 = Policy defined by remote FW  
2 = Policy pushed CPP  
4 = Policy from server |
| IPsec-Sec-Association                | Y        |     |     | String      | Single                 | Name of the security association |
| IPsec-Split-DNS-Names                | Y        | Y   | Y   | String      | Single                 | Specifies the list of secondary domain names to send to the client (1 - 255 characters). |
| IPsec-Split-Tunneling-Policy         | Y        | Y   | Y   | Integer     | Single                 | 0 = Tunnel everything  
1 = Split tunneling  
2 = Local LAN permitted |
| IPsec-Split-Tunnel-List              | Y        | Y   | Y   | String      | Single                 | Specifies the name of the network or access list that describes the split tunnel inclusion list. |
| IPsec-Tunnel-Type                    | Y        | Y   | Y   | Integer     | Single                 | 1 = LAN-to-LAN  
2 = Remote access |
| IPsec-User-Group-Lock                | Y        |     |     | Boolean     | Single                 | 0 = Disabled  
1 = Enabled |
| L2TP-Encryption                      | Y        |     |     | Integer     | Single                 | Bitmap:  
1 = Encryption required  
2 = 40 bit  
4 = 128 bits  
8 = Stateless-Req  
15 = 40/128-Encr/Stateless-Req |
| L2TP-MPPC-Compression                | Y        |     |     | Integer     | Single                 | 0 = Disabled  
1 = Enabled |
| MS-Client-Subnet-Mask                | Y        | Y   | Y   | String      | Single                 | An IP address |
| PFS-Required                         | Y        | Y   | Y   | Boolean     | Single                 | 0 = No  
1 = Yes |
| Port-Forwarding-Name                 | Y        | Y   |     | String      | Single                 | Name string (for example, “Corporate-Apps”) |
| PPTP-Encryption                      | Y        |     |     | Integer     | Single                 | Bitmap:  
1 = Encryption required  
2 = 40 bit  
4 = 128 bits  
8 = Stateless-Required  
Example:  
15 = 40/128-Encr/Stateless-Req |
| PPTP-MPPC-Compression                | Y        |     |     | Integer     | Single                 | 0 = Disabled  
1 = Enabled |
### Configuring an External Server for Authorization and Authentication

#### Appendix C

---

#### Configuring an External LDAP Server

<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>
</tbody>
</table>
| Required-Client-Firewall-Vendor-Code  | Y        | Y   | Y   | Integer     | Single                 | 1 = Cisco Systems (with Cisco Integrated Client)  
|                                       |          |     |     |             |                        | 2 = Zone Labs    
|                                       |          |     |     |             |                        | 3 = NetworkICE   
|                                       |          |     |     |             |                        | 4 = Sygate       
|                                       |          |     |     |             |                        | 5 = Cisco Systems (with Cisco Intrusion Prevention Security Agent) |
| Required-Client-Firewall-Description  | Y        | Y   | Y   | String      | Single                 | —               |
| Required-Client-Firewall-Product-Code | Y        | Y   | Y   | 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  |
| Require-HW-Client-Auth                | Y        | Y   | Y   | Boolean     | Single                 | 0 = Disabled  
|                                       |          |     |     |             |                        | 1 = Enabled     |
| Require-Individual-User-Auth          | Y        | Y   | Y   | Integer     | Single                 | 0 = Disabled  
|                                       |          |     |     |             |                        | 1 = Enabled     |
| Secondary-DNS                         | Y        | Y   | Y   | String      | Single                 | An IP address |
| Secondary-WINS                        | Y        | Y   | Y   | String      | Single                 | An IP address |
| SEP-Card-Assignment                  |          |     |     | Integer     | Single                 | Not used       |
| Simultaneous-Logins                   | Y        | Y   | Y   | Integer     | Single                 | 0 - 2147483647 |
| Strip-Realm                           | Y        | Y   | Y   | Boolean     | Single                 | 0 = Disabled  
|                                       |          |     |     |             |                        | 1 = Enabled     |
| TACACS-AuthType                       | Y        | Y   | Y   | Integer     | Single                 | —              |
| TACACS-Privilege-Level                | Y        | Y   | Y   | Integer     | Single                 | —              |
| Tunnel-Group-Lock                     | Y        | Y   |     | String      | Single                 | Name of the tunnel group or “none” |
### Table C-2  ASA 1000V 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 ASA 1000V Supported Cisco Attributes for LDAP Authorization (continued)

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>VPN 3000</th>
<th>ASA</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>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>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>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>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></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>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>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>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>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>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>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>Integer</td>
<td>Single</td>
<td>0 = None 1 = SSL 2 = New tunnel 3 = Any (sets to SSL)</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:inacl#1= for standard access lists or webvpn:inacl# = 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>

**Table C-2 ASA 1000V 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 = 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 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 1 = Enabled</td>
</tr>
<tr>
<td>WebVPN-URL-List</td>
<td>Y</td>
<td></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:inacl#1= for standard access lists or webvpn:inacl# = 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.

<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 10.159.2.0 0.0.0.255 log</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</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>URL Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>any</td>
<td>All URLs</td>
</tr>
<tr>
<td>http://</td>
<td>Post://</td>
</tr>
<tr>
<td>https://</td>
<td>Ssh://</td>
</tr>
<tr>
<td>cifs://</td>
<td>Ica://</td>
</tr>
<tr>
<td>rdp://</td>
<td>Telnet://</td>
</tr>
<tr>
<td>citrix://</td>
<td>Imap4://</td>
</tr>
<tr>
<td>rdp2://</td>
<td>Vnc://</td>
</tr>
<tr>
<td>citrixs://</td>
<td>Ftp://</td>
</tr>
<tr>
<td>smart-tunnel://</td>
<td></td>
</tr>
<tr>
<td>http://</td>
<td>Pop3://</td>
</tr>
<tr>
<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 ASA 1000V is the source.

**Table C-5** lists the tokens for the Cisco-AV-pair attribute:

**Table C-5 ASA 1000V-Supported Tokens**

<table>
<thead>
<tr>
<th>Token</th>
<th>Syntax Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip:inacl#Num=</td>
<td>N/A (Identifier)</td>
<td>(Where Num is a unique integer.) Starts all AV pair access control lists.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Enforces access lists for remote IPsec and SSL VPN (SVC) tunnels.</td>
</tr>
<tr>
<td>webvpn:inacl#Num=</td>
<td>N/A (Identifier)</td>
<td>(Where Num is a unique integer.) Starts all clientless SSL AV pair access</td>
</tr>
<tr>
<td></td>
<td></td>
<td>control lists. Enforces access lists for clientless (browser-mode) tunnels.</td>
</tr>
<tr>
<td>deny</td>
<td>Action</td>
<td>Denies action. (Default)</td>
</tr>
<tr>
<td>permit</td>
<td>Action</td>
<td>Allows action.</td>
</tr>
<tr>
<td>icmp</td>
<td>Protocol</td>
<td>Internet Control Message Protocol (ICMP)</td>
</tr>
<tr>
<td>1</td>
<td>Protocol</td>
<td>Internet Control Message Protocol (ICMP)</td>
</tr>
<tr>
<td>IP</td>
<td>Protocol</td>
<td>Internet Protocol (IP)</td>
</tr>
<tr>
<td>0</td>
<td>Protocol</td>
<td>Internet Protocol (IP)</td>
</tr>
<tr>
<td>TCP</td>
<td>Protocol</td>
<td>Transmission Control Protocol (TCP)</td>
</tr>
<tr>
<td>6</td>
<td>Protocol</td>
<td>Transmission Control Protocol (TCP)</td>
</tr>
<tr>
<td>UDP</td>
<td>Protocol</td>
<td>User Datagram Protocol (UDP)</td>
</tr>
<tr>
<td>17</td>
<td>Protocol</td>
<td>User Datagram Protocol (UDP)</td>
</tr>
<tr>
<td>any</td>
<td>Hostname</td>
<td>Rule applies to any host.</td>
</tr>
<tr>
<td>host</td>
<td>Hostname</td>
<td>Any alpha-numeric string that denotes a hostname.</td>
</tr>
<tr>
<td>log</td>
<td>Log</td>
<td>When the event occurs, a filter log message appears. (Same as permit and log or deny and log.)</td>
</tr>
<tr>
<td>lt</td>
<td>Operator</td>
<td>Less than value</td>
</tr>
</tbody>
</table>
Appendix C Configuring an External Server for Authorization and Authentication

Configuring an External LDAP Server

This section presents example procedures for configuring authentication and authorization on the ASA 1000V 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 ASA 1000V 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 ASA 1000V, create an attribute map that maps physicalDeliveryOfficeName to the Cisco attribute Banner1. During authentication, the ASA 1000V 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*

Step 3 Create an LDAP attribute map on the ASA 1000V.

The following example creates the map Banner and maps the AD/LDAP attribute physicalDeliveryOfficeName to the Cisco attribute Banner1:

```
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:

```
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 ASA 1000V. 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 ASA 1000V 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 ASA 1000V 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-1)#
```

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 ASA 1000V 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 ASA 1000V 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.

**Figure C-6 Assign Static IP Address**

**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 ASA 1000V 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 ASA 1000V (see Figure C-8).

*Figure C-7 Verify the Banner for the AnyConnect Session*
Step 7  Use the `show vpn-sessiondb svc` command to view the session details and verify the address assigned:

```
hostname# show vpn-sessiondb svc
```

```
Session Type: SVC
Username     : web1                   Index        : 31
Assigned IP  : 10.1.1.2              Public IP    : 10.86.181.70
Protocol     : Clientless SSL-Tunnel DTLS-Tunnel
Encryption   : RC4 AES128             Hashing      : SHA1
Bytes Tx     : 304140                 Bytes Rx     : 470506
Group Policy : VPN_User_Group         Tunnel Group : Group1_TunnelGroup
Login Time   : 11:13:05 UTC Tue Aug 28 2007
Duration     : 0h:01m:48s
NAC Result   : Unknown
VLAN Mapping : N/A                    VLAN         : none
```

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 C-6  Bitmap Values for Cisco Tunneling-Protocol Attribute

<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>4(^1)</td>
<td>IPsec (IKEv1)</td>
</tr>
<tr>
<td>8(^2)</td>
<td>L2TP/IPsec</td>
</tr>
</tbody>
</table>
```

\(^1\) These values are supported on ASA 5500 models.

\(^2\) These values are supported on ASA 5580 models.
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 ASA 1000V 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 Server for Authorization and Authentication

Appendix C

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 ASA 1000V.

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 ASA 1000V to map that attribute to the Cisco attribute Access-Hours. During authentication, the ASA 1000V 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.

```plaintext
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:

```plaintext
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:

```plaintext
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
- ASA 1000V RADIUS Authorization Attributes, page C-27
- ASA 1000V IETF RADIUS Authorization Attributes, page C-36

Reviewing the RADIUS Configuration Procedure

This section describes the RADIUS configuration steps required to support authentication and authorization of ASA 1000V users.

To set up the RADIUS server to interoperate with the ASA 1000V, preform the following steps:

**Step 1** Load the ASA 1000V 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 ASA 1000V attributes. Obtain this dictionary file, cisco3k.dct, from the Cisco Download Software Center on Cisco.com or from the ASA 1000V 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 ASA 1000V attribute. To define an attribute, use the attribute name or number, type, value, and vendor code (3076). For a list of ASA 1000V 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.

ASA 1000V 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 ASA 1000V 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 ASA 1000Vs 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 ASA 1000V except for the following attribute numbers: 146, 150, 151, and 152. These attribute numbers are upstream attributes that are sent from the ASA 1000V 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>
<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>
<tr>
<td>Tunneling-Protocols</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>11</td>
<td>Integer</td>
<td>Single</td>
<td>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).</td>
</tr>
<tr>
<td>IPsec-Sec-Association</td>
<td></td>
<td></td>
<td></td>
<td>12</td>
<td>String</td>
<td>Single</td>
<td>Name of the security association</td>
</tr>
<tr>
<td>IPsec-Authentication</td>
<td></td>
<td></td>
<td></td>
<td>13</td>
<td>Integer</td>
<td>Single</td>
<td>0 = None 1 = RADIUS 2 = LDAP (authorization only) 3 = NT Domain 4 = SDI 5 = Internal 6 = RADIUS with Expiry 7 = Kerberos/Active Directory</td>
</tr>
<tr>
<td>Banner1</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>15</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</td>
</tr>
<tr>
<td>IPsec-Allow-Passwd-Store</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>16</td>
<td>Boolean</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>Attribute Name</td>
<td>VPN 3000</td>
<td>ASA</td>
<td>PIX</td>
<td>Attr. No.</td>
<td>Syntax/Type</td>
<td>Single or Multi-Valued</td>
<td>Description or Value</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>----------</td>
<td>-----</td>
<td>-----</td>
<td>-----------</td>
<td>-------------</td>
<td>------------------------</td>
<td>----------------------------------------------------------------------------------------</td>
</tr>
<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 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 ASA 1000V 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</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1 = Enabled</td>
<td></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</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1 = Enabled</td>
<td></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</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1 = Enabled</td>
<td></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</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1 = Enabled</td>
<td></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</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2 = If supported by peer certificate</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3 = Do not check</td>
<td></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</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1 = Enabled</td>
<td></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</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1 = Enabled</td>
<td></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)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2 = Zone Labs</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3 = NetworkICE</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4 = Sygate</td>
<td></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>
<td></td>
</tr>
</tbody>
</table>
### Table C-7 ASA 1000V 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>Required-Client-Firewall-Product-Code</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y 46</td>
<td>Integer</td>
<td>Single</td>
<td>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</td>
</tr>
<tr>
<td>Required-Client-Firewall-Description</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y 47</td>
<td>String</td>
<td>Single</td>
<td>String</td>
</tr>
<tr>
<td>Require-HW-Client-Auth</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y 48</td>
<td>Boolean</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>Required-Individual-User-Auth</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y 49</td>
<td>Integer</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>Y 50</td>
<td>Integer</td>
<td>Single</td>
<td>1-35791394 minutes</td>
</tr>
<tr>
<td>Cisco-IP-Phone-Bypass</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y 51</td>
<td>Integer</td>
<td>Single</td>
<td>0 = No split tunneling 1 = Split tunneling 2 = Local LAN permitted</td>
</tr>
<tr>
<td>IPsec-Split-Tunneling-Policy</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y 55</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-Required-Client-Firewall-Capability</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y 56</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-Client-Firewall-Filter-Name</td>
<td>Y</td>
<td></td>
<td></td>
<td>Y 57</td>
<td>String</td>
<td>Single</td>
<td>Specifies the name of the filter to be pushed to the client as firewall policy</td>
</tr>
<tr>
<td>IPsec-Client-Firewall-Filter-Optional</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y 58</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Required 1 = Optional</td>
</tr>
<tr>
<td>IPsec-Backup-Servers</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y 59</td>
<td>String</td>
<td>Single</td>
<td>1 = Use Client-Configured list 2 = Disable and clear client list 3 = Use Backup Server list</td>
</tr>
<tr>
<td>IPsec-Backup-Server-List</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y 60</td>
<td>String</td>
<td>Single</td>
<td>Server Addresses (space delimited)</td>
</tr>
</tbody>
</table>
Table C-7   ASA 1000V 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 = None 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></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></td>
<td></td>
<td></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, 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></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></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></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></td>
<td></td>
<td></td>
<td>75</td>
<td>Integer</td>
<td>Single</td>
<td>0 = Disabled 1 = Enabled</td>
</tr>
<tr>
<td>WebVPN-Homepage</td>
<td></td>
<td></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></td>
<td></td>
<td></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></td>
<td></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></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></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 ASA 1000V 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-Keppalive-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 Y Y</td>
<td></td>
<td></td>
<td>85</td>
<td>String</td>
<td>Single</td>
<td>Name of the tunnel group or &quot;none&quot;</td>
</tr>
<tr>
<td>Access-List-Inbound</td>
<td>Y Y</td>
<td></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 Y</td>
<td></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>Access list</td>
</tr>
<tr>
<td>WebVPN-URL-Entry-Enable</td>
<td>Y Y</td>
<td></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-Over Outlook Exchange-Proxy</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 Y</td>
<td></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  ASA 1000V 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</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 1 = 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</td>
</tr>
</tbody>
</table>

Cisco ASA 1000V CLI Configuration Guide for ASDM Mode
### Table C-7  ASA 1000V 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</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>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)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2 = AnyConnect Client SSL VPN</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3 = Clientless SSL VPN</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4 = Cut-Through-Proxy</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>5 = L2TP/IPsec SSL VPN</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>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</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1 = AnyConnect Client SSL VPN</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2 = AnyConnect Client IPsec VPN (IKEv2)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3 = Clientless SSL VPN</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4 = Clientless Email Proxy</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>5 = Cisco VPN Client (IKEv1)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>6 = IKEv1 LAN-LAN</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>7 = IKEv2 LAN-LAN</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>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</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1 = Clientless</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2 = Client</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3 = Client Only</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>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  ASA 1000V 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>Y</td>
<td>220</td>
<td>Integer</td>
<td>Single</td>
<td>An integer between 0 and 15.</td>
</tr>
</tbody>
</table>

ASA 1000V IETF RADIUS Authorization Attributes

Table C-8 lists the supported IETF RADIUS attributes.

Table C-8  ASA 1000V 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>
</table>
| IETF-Radius-Class      | Y        | Y   | Y   | 25        | String      | Single                 | 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:  
  * group policy name  
  * OU=group policy name  
  * OU=group policy name |
| IETF-Radius-Filter-Id  | Y        | Y   | Y   | 11        | String      | Single                 | Access list name that is defined on the ASA 1000V, which applies only to full tunnel IPsec and SSL VPN clients |
| IETF-Radius-Framed-IP-Address | Y   | Y   | Y   | n/a       | String      | Single                 | An IP address                                                                    |
| IETF-Radius-Framed-IP-Netmask | Y   | Y   | Y   | n/a       | String      | Single                 | An IP address mask                                                              |
| IETF-Radius-Idle-Timeout | Y   | Y   | Y   | 28        | Integer     | Single                 | Seconds                                                                         |
Configuring an External TACACS+ Server

The ASA 1000V 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-8 ASA 1000V Supported IETF RADIUS Attributes and Values

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IETF-Radius-Service-Type</td>
<td>IETF-RADIUS-Service-Type Y Y Y 6 Integer Single Seconds. Possible Service Type values:</td>
</tr>
<tr>
<td></td>
<td>.Administrative—User is allowed access to configure prompt.</td>
</tr>
<tr>
<td></td>
<td>.NAS-Prompt—User is allowed access to exec prompt.</td>
</tr>
<tr>
<td></td>
<td>.remote-access—User is allowed network access</td>
</tr>
<tr>
<td>IETF-Radius-Session-Timeout</td>
<td>Y Y Y 27 Integer Single Seconds</td>
</tr>
</tbody>
</table>

### 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>
</tbody>
</table>
### Table C-10  Supported TACACS+ Accounting Attributes (continued)

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<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>
<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.

**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 ASA 1000V 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.

**A record address**
“A” stands for address, and refers to name-to-address mapped records in DNS.
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 ASA 1000V 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 ASA 1000V.

ASDM Adaptive Security Device Manager. An application for managing and configuring a single ASA 1000V.

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-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

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 ASA 1000V 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.

C

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 ASA 1000V.

**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 ASA 1000V that represents the equivalent of settings, preferences, and properties administered by ASDM or the CLI.

**cookie**

A cookie is an object stored by a browser. Cookies contain information, such as user preferences, to persistent storage.

**CPU**

Central Processing Unit. Main processor.

**CRC**

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.
CRL  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 CA, certificate, public key, RA.

CRV  Call Reference Value. Used by H.225.0 to distinguish call legs signaled between two entities.

cryptography  Encryption, authentication, integrity, keys and other services used for secure communication over networks. See also VPN and IPsec.

crypto map  A data structure with a unique name and sequence number that is used for configuring VPNs on the ASA 1000V. 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 IKE and IPsec. See also VPN.

CTIQBE  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 NAT, PAT, and bidirectional NAT. 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 ASA 1000V.

cut-through proxy  Enables the ASA 1000V 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.

D

data confidentiality  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.

data integrity  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.

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 ASA 1000V 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.
E

ECHO
See ping, ICMP. See also inspection engine.

EGP
Exterior Gateway Protocol. Replaced by BGP. The ASA 1000V does not support EGP. See also BGP.

EIGRP
Enhanced Interior Gateway Routing Protocol. The ASA 1000V 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.

F

Fixup
See inspection engine.

Flash, Flash memory
A nonvolatile storage device used to store the configuration file when the ASA 1000V 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 ASA 1000V.

FTP
File Transfer Protocol. Part of the TCP/IP protocol stack, used for transferring files between hosts.

G

global configuration mode
Global configuration mode lets you change the ASA 1000V 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.
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.

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-56 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 ASA 1000V 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.

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 ASA 1000V 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 ASA 1000V 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 ASA 1000V. See also interface, interface name.
inspection engine
The ASA 1000V 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 ASA 1000V can inspect are CTIQBE, FTP, H.323, HTTP, MGCP, SMTP, and SNMP.

interface
The physical connection between a particular network and a ASA 1000V.

interface IP address
The IP address of the ASA 1000V 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 ASA 1000V 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.

IP address
An IP protocol address. A ASA 1000V 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 tunneling 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.
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.

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 ASA 1000V features in a manner similar to Cisco IOS software Modular QoS CLI.

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.

N
N2H2
A third-party, policy-oriented filtering application that works with the ASA 1000V 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 ASA 1000V 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 ASA 1000V 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.

NTLM

NTP
Network Time Protocol.

O

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 ASA 1000V does not support 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 ASA 1000V; 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 ASA 1000V 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.

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 ASA 1000V s 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 ASA 1000V s 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.
**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 ASA 1000V 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 ASA 1000V configuration as part of defining a security policy by their literal values or port numbers. Possible ASA 1000V 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 ASA 1000V 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.
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 ASA 1000V 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.

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.
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 ASA 1000V configuration to define security policy for a particular situation. See also ACE, ACL, NAT.

running configuration  The configuration currently running in RAM on the ASA 1000V. The configuration that determines the operational characteristics of the ASA 1000V.

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.

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 ASA 1000V 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 profile interface  An interface used only on the ASA 1000V to apply policies.

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).
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 ASA 1000V 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.

SMTP
Simple Mail Transfer Protocol. SMTP is an Internet protocol that supports email services.

SNMP

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 ASA 1000V 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.

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.

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 ASA 1000V 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


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 ASA 1000V responds on behalf of the server with an empty SYN/ACK segment. The ASA 1000V 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 ASA 1000V 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 ASA 1000V 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.
### Glossary

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>TLS</strong></td>
<td>Transport Layer Security. A future IETF protocol to replace SSL.</td>
</tr>
<tr>
<td><strong>traffic policing</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>transform set</strong></td>
<td>See IPsec transform set.</td>
</tr>
<tr>
<td><strong>translate, translation</strong></td>
<td>See xlate.</td>
</tr>
<tr>
<td><strong>transport mode</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>TSP</strong></td>
<td>TAPI Service Provider. See also TAPI.</td>
</tr>
<tr>
<td><strong>tunnel mode</strong></td>
<td>An IPsec encryption mode that encrypts both the header and data portion (payload) of each packet. Tunnel mode is more secure than transport mode.</td>
</tr>
<tr>
<td><strong>tunnel</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>Turbo ACL</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>U</strong></td>
<td><strong>UDP</strong> 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.</td>
</tr>
<tr>
<td><strong>Unicast RPF</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>user EXEC mode</strong></td>
<td>The lowest privilege level at the ASA CLI. The user EXEC mode prompt appears as follows when you first access the ASA 1000V: hostname&gt; See also command-specific configuration mode, global configuration mode, and privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>UTC</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>UUIE</strong></td>
<td>User-User Information Element. An element of an H.225 packet that identifies the users implicated in the message.</td>
</tr>
</tbody>
</table>
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 ASA 1000V.

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.

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  18-1
authentication
CLI access  18-21
privileged EXEC mode  18-21
authorization
command  18-24
local database support  18-8
server  29-4
adding  18-11
types  18-1
support summary  18-5
abbreviating commands  A-3
access lists
about  9-1
ACE logging, configuring  10-1
deny flows, managing  10-4
global access rules  16-2
implicit deny  9-2, 16-2
inbound  16-3
IP address guidelines  9-2
IPsec  26-18
logging  10-1
NAT guidelines  9-2

object groups  8-2
outbound  16-3
remarks  9-6
scheduling activation  8-14
ACEs
See access lists
Active/Standby failover
about  3-2
actions  3-5
command replication  3-3
configuration synchronization  3-3
device initialization  3-3
primary unit  3-2
secondary unit  3-2
triggers  3-4
Active Directory procedures  C-16 to ??
Adaptive Security Algorithm  1-10
administrative access
using ICMP for  17-9
administrative distance  7-6, 7-7
Advanced Encryption Standard (AES)  26-3, 26-5
AIP SSM
port-forwarding
enabling  4-6
alternate address, ICMP message  8-9
analyzing syslog messages  29-2
application inspection
about  19-1
applying  19-5
configuring  19-5
inspection class map  15-4
inspection policy map  15-2
special actions  15-1
ARP
  NAT  11-21
ARP test, failover  3-13
ASA (Adaptive Security Algorithm)  1-10
ASDM software
  allowing access  17-5
  installing  32-2
asymmetric routing
  TCP state bypass  24-4
attacks
  DNS HINFO request  25-7
  DNS request for all records  25-7
  DNS zone transfer  25-7
  DNS zone transfer from high port  25-7
  fragmented ICMP traffic  25-6
  IP fragment  25-4
  IP impossible packet  25-4
  large ICMP traffic  25-6
  ping of death  25-6
  proxied RPC request  25-7
  statd buffer overflow  25-8
  TCP FIN only flags  25-7
  TCP NULL flags  25-6
  TCP SYN+FIN flags  25-6
  UDP bomb  25-7
  UDP chargen DoS  25-7
  UDP snort  25-7
attributes
  RADIUS  C-27
attribute-value pairs
  TACACS+  C-37
attribute-value pairs (AVP)  27-7
authentication
  about  18-2
  CLI access  18-21
  privileged EXEC mode  18-21
authorization
  about  18-2
  command  18-24
Auto-Update, configuring  32-7

B
backup server attributes, group policy  27-15
bits subnet masks  B-3
broadcast Ping test  3-13
building blocks  8-1
bypassing firewall checks  24-3

capturing packets  33-11
cascading access lists  26-13
changing the severity level  29-17
Cisco-AV-Pair LDAP attributes  C-13
Cisco IP Phones, application inspection  21-25
Class A, B, and C addresses  B-1
class-default class map  14-8
classes, logging
  filtering messages by  29-15
  message class variables  29-3
types  29-3
class map
  inspection  15-4
Layer 3/4
  management traffic  14-13
  match commands  14-11, 14-14
  through traffic  14-11
regular expression  8-13
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  18-3
   configuring  18-24
command prompts  A-2
comments
   configuration  A-5
configuration
   clearing  2-9
   comments  A-5
   saving  2-7
   text file  2-9
   viewing  2-8
configuration examples
logging  29-18
configuration examples for SNMP  30-17
configuration mode
   accessing  2-3
   prompt  A-2
connection blocking  25-2
connection limits
   configuring  24-1
console port logging  29-10
conversion error, ICMP message  B-9
Coredump  33-11
   crash dump  33-11
creating a custom event list  29-12
crypto map
   access lists  26-18
   applying to interfaces  26-18, 28-10
   clearing configurations  26-26
   definition  26-11
   dynamic  26-23
   entries  26-11
   examples  26-19
   policy  26-11
crypto show commands table  26-25
   custom messages list
      logging output destination  29-4

D
date and time in messages  29-16
debug messages  33-10
default
   DefaultL2Lgroup  27-1
   DefaultRAgroup  27-1
   domain name, group policy  27-13
   group policy  27-1, 27-7
   LAN-to-LAN tunnel group  27-4
   routes, defining equal cost routes  7-5
   tunnel group  26-9, 27-2
default policy  14-6
default routes
   about  7-4
   configuring  7-4
deleting files from Flash  32-2
deny flows, logging  10-4
deny in a crypto map  26-13
DES, IKE policy keywords (table)  26-3, 26-5
device ID, including in messages  29-16
device ID in messages  29-16
DfltGrpPolicy  27-7
DHCP
   options  6-3
   relay  6-5
   server  6-2
   DHCP Intercept, configuring  27-14
   DHCP services  5-6
   Diffie-Hellman
      Group 5  26-4, 26-5
      groups supported  26-4, 26-5
directory hierarchy search  C-3
disabling messages  29-17
disabling messages, specific message IDs  29-17
DMZ, definition  1-8
DNS
   inspection
      about  20-2
managing 20-1
rewrite, about 20-2
rewrite, configuring 20-3
NAT effect on 11-22
server, configuring 5-10
DNS HINFO request attack 25-7
DNS request for all records attack 25-7
DNS zone transfer attack 25-7
DNS zone transfer from high port attack 25-7
domain attributes, group policy 27-13
domain name 5-3
dotted decimal subnet masks B-3
dual-ISP support 7-8
duplex, configuring 4-5
dynamic crypto map 26-23
See also crypto map
dynamic NAT
about 11-9
network object NAT 12-3	
twice NAT 13-3
dynamic PAT
network object NAT 12-5
See also NAT
twice NAT 13-8
for management traffic 9-3
external group policy, configuring 27-9

F
facility, syslog 29-8
failover
about 3-1
Active/Standby, See Active/Standby failover
configuration file
terminal messages, Active/Standby 3-3
debug messages 3-14
disabling 3-23
Ethernet failover cable 1-9, 3-6
failover link 3-6
forcing 3-23
guidelines 30-8
health monitoring 3-12
interface health 3-13
interface monitoring 3-13
interface tests 3-13
link communications 3-6
MAC addresses
about 3-2
monitoring, health 3-12
network tests 3-13
primary unit 3-2
restoring a failed group 3-23
restoring a failed unit 3-23
secondary unit 3-2
SNMP syslog traps 3-14
Stateful Failover, See Stateful Failover
state link 3-7
system log messages 3-14
testing 3-23
unit health 3-13
fast path 1-10
Fibre Channel interfaces
default settings 16-4

E
echo reply, ICMP message B-9
ECMP 7-7
editing command lines A-3
enable command 2-2
enabling logging 29-5
enabling secure logging 29-15
Ethernet
duplex 4-5
MTU 4-8
speed 4-5
extended ACLs
configuring
Cisco ASA 1000V CLI Configuration Guide for ASDM Mode
filtering
  show command output A-4
filtering messages 29-3
Flash memory
  removing files 32-2
flash memory available for logs 29-14
format of messages 29-2
fragmentation policy, IPsec 26-8
fragmented ICMP traffic attack 25-6
fragment protection 1-9
fragment size 25-2
FTP inspection
  about 20-11
  configuring 20-11

G

general attributes, tunnel group 27-2
general parameters, tunnel group 27-2
general tunnel-group connection parameters 27-2
global IPsec SA lifetimes, changing 26-20
group policy
  attributes 27-10
  backup server attributes 27-15
  configuring 27-9
default domain name for tunneled packets 27-13
definition 27-1, 27-7
domain attributes 27-13
external, configuring 27-9
internal, configuring 27-10
split-tunneling domains 27-14
VPN attributes 27-10

group policy, default 27-7
  SNMP 30-7

I

ICMP
  rules for access to ASDM 17-9
testing connectivity 33-1
type numbers B-9

identity NAT
  about 11-12
  network object NAT 12-11
twice NAT 13-20

ID method for ISAKMP peers, determining 26-7

IKE

H.225 timeouts 21-9
H.245 troubleshooting 21-10
H.323 inspection
  about 21-4
  configuring 21-3
  limitations 21-5
troubleshooting 21-10
hairpinning 26-18
help, command line A-4
high availability
  about 3-1
HMAC hashing method 26-2, 28-3
host
  SNMP 30-7
hostname
  configuring 5-3
hosts, subnet masks for B-3
HTTP(S)
  authentication 18-21
HTTP inspection
  configuring 20-16
HTTPS/Telnet/SSH
  allowing network or host access to ASDM 17-1
hub-and-spoke VPN scenario 26-18
benefits 26-2, 28-3
creating policies 26-5
See also ISAKMP
IKEv1 26-10
ILS inspection 22-1
IM 21-19
implementing SNMP 30-7
inbound access lists 16-3
information reply, ICMP message B-9
information request, ICMP message B-9
inheritance
tunnel group 27-1
inside, definition 1-8
inspection_default class-map 14-8
inspection engines
See application inspection
Instant Messaging inspection 21-19
intercept DHCP, configuring 27-14
interface
MTU 4-8
interfaces
default settings 16-4
duplex 4-5
enabling 4-8
failover monitoring 3-13
IP address 4-7
MAC addresses
manually assigning to interfaces 4-8
naming, physical and subinterface 4-7
speed 4-5
internal group policy, configuring 27-10
IP addresses
classes B-1
interface 4-7
private B-2
subnet mask B-4
IP fragment attack 25-4
IP impossible packet attack 25-4
IP overlapping fragments attack 25-5
IPsec
access list 26-18
basic configuration with static crypto maps 26-21
Cisco VPN Client 26-2
configuring 26-1, 26-9
crypto map entries 26-11
fragmentation policy 26-8
over NAT-T, enabling 26-8
SA lifetimes, changing 26-20
tunnel 26-10
view configuration commands table 26-25
IP spoofing, preventing 25-1
IP teardrop attack 25-5
ISAKMP
about 26-2
configuring 26-1
determining an ID method for peers 26-7
disabling in aggressive mode 26-7
K
Kerberos
configuring 18-11
support 18-7
L
LAN-to-LAN tunnel group, configuring 27-4
large ICMP traffic attack 25-6
Layer 3/4
matching multiple policy maps 14-5
LDAP
application inspection 22-1
attribute mapping 18-15, 18-16
Cisco-AV-pair C-13
configuring 18-11
configuring a AAA server C-2 to ??
directory search C-3
example configuration procedures  C-16 to ??

hierarchy example  C-3

SASL  18-7

user authentication  18-7

link up/down test  3-13

local user database
adding a user  18-17
configuring  18-17
logging in  18-22
support  18-8

lockout recovery  18-33

logging
access lists  10-1

classes
filtering messages by  29-4
types  29-3, 29-15

device-id, including in system log messages  29-16

e-mail
source address  29-9

EMBLEM format  29-13

facility option  29-8

filtering
by message class  29-15
by message list  29-4
by severity level  29-1

logging queue, configuring  29-14

output destinations  29-7
console port  29-7, 29-9, 29-10
internal buffer  29-1, 29-6
Telnet or SSH session  29-6

queue
changing the size of  29-14
configuring  29-14
viewing queue statistics  29-18

severity level, changing  29-18
timestamp, including  29-16

logging feature history  29-19

logging queue
configuring  29-14

login
banner, configuring  17-6
enable  2-2
global configuration mode  2-3
local user  18-22
password  5-2
SSH  17-4
Telnet  5-2

M

MAC addresses
failover  3-2
manually assigning to interfaces  4-8

management interfaces
default settings  16-4

mapped addresses
guidelines  11-20

mask
reply, ICMP message  B-9
request, ICMP message  B-9

Master Passphrase  5-6

match commands
inspection class map  15-4
Layer 3/4 class map  14-11, 14-14

MD5, IKE policy keywords (table)  26-4, 26-5

message filtering  29-3

message list
filtering by  29-4
message-of-the-day banner  17-6

messages, logging
classes
about  29-4
list of  29-3

component descriptions  29-2

filtering by message list  29-4

format of  29-2

message list, creating  29-12

severity levels  29-3
messages classes 29-3
messages in EMBLEM format 29-13
metacharacters, regular expression 8-11
MGCP inspection
   about 21-11
   configuring 21-11
mgmt0 interfaces
   default settings 16-4
MIBs 30-3
MIBs for SNMP 30-18
mobile redirection, ICMP message B-9
monitoring
   failover 3-12
   SNMP 30-1
monitoring logging 29-18
More prompt A-5
MPF
   default policy 14-6
   examples 14-17
   features 14-2
   flows 14-5
   matching multiple policy maps 14-5
   service policy, applying 14-16
   See also policy map
MTU 4-8

N
naming an interface
   other models 4-7
NAT
   about 11-1
   bidirectional initiation 11-3
   disabling proxy ARP for global addresses 7-7
DNS 11-22
   dynamic
      about 11-9
   dynamic NAT
      network object NAT 12-3
twice NAT 13-3
dynamic PAT
   about 11-11
   network object NAT 12-5
twice NAT 13-8
identity
   about 11-12
identity NAT
   network object NAT 12-11
twice NAT 13-20
implementation 11-14
interfaces 11-19
mapped address guidelines 11-20
network object
   comparison with twice NAT 11-14
network object NAT
   about 11-15
   configuring 12-1
   dynamic NAT 12-3
dynamic PAT 12-5
   examples 12-13
guidelines 12-2
   identity NAT 12-11
   monitoring 12-13
   prerequisites 12-2
   static NAT 12-9
no proxy ARP 12-12, 13-19
object
   extended PAT 12-5
   flat range for PAT 12-5
   route lookup 12-12, 13-24
RPC not supported with 22-3
rule order 11-18
   static
      about 11-4
      few-to-many mapping 11-8
      many-to-few mapping 11-7, 11-8
      one-to-many 11-7
      static NAT
network object NAT 12-9

twice NAT 13-15
static with port translation
about 11-4
terminology 11-3
twice
extended PAT 13-9
flat range for PAT 13-8
twice NAT
about 11-15
comparison with network object NAT 11-14
configuring 13-1
dynamic NAT 13-3
dynamic PAT 13-8
examples 13-24
guidelines 13-2
identity NAT 13-20
monitoring 13-24
prerequisites 13-2
static NAT 13-15
types 11-3
VPN 11-13
VPN client rules 11-18

NAT-T
enabling IPsec over NAT-T 26-8
using 26-8
Network Activity test 3-13

network object NAT
about 11-15
comparison with twice NAT 11-14
configuring 12-1
dynamic NAT 12-3
dynamic PAT 12-5
examples 12-13
guidelines 12-2
identity NAT 12-11
monitoring 12-13
prerequisites 12-2
static NAT 12-9

no proxy ARP 13-19
NTLM support 18-7
NT server
configuring 18-11
support 18-7

O

object groups
about 8-1
configuring 8-5
removing 8-10

object NAT
See network object NAT

open ports 8-8
outbound access lists 16-3
output destination 29-4
output destinations 29-1, 29-6
e-mail address 29-1, 29-6
SNMP management station 29-1, 29-6
Telnet or SSH session 29-1, 29-6
outside, definition 1-8

P

packet
capture 33-11
packet trace, enabling 33-7
paging screen displays 4-5
parameter problem, ICMP message 4-9
passwords
changing 5-2
recovery 33-8
security appliance 5-2

PAT
See dynamic PAT
peers
alerting before disconnecting 26-9
ISAKMP, determining ID method 26-7
permit in a crypto map 26-13
ping
See ICMP
ping of death attack 25-6
policy map
  inspection 15-2
  Layer 3/4
    about 14-1
    flows 14-5
pools, address
  DHCP 6-2
port-forwarding
  enabling 4-6
ports
  open on device B-8
  TCP and UDP B-5
port translation
  about 11-4
primary unit, failover 3-2
private networks B-2
privileged EXEC mode, accessing 2-2
privileged mode
  accessing 2-2
  prompt A-2
prompts
  command A-2
  more A-5
protocol numbers and literal values B-5
proxied RPC request attack 25-7
proxy ARP
  NAT
    proxy ARP 11-21
proxy ARP, disabling 7-7
proxy servers
  SIP and 21-19
question mark
  command string A-4
  help A-4
queue, logging
  changing the size of 29-14
  viewing statistics 29-18
RADIUS
  attributes C-27
  Cisco AV pair C-13
  configuring a AAA server C-27
  configuring a server 18-11
  support 18-5
RAS, H.323 troubleshooting 21-10
rate limit 29-17
RealPlayer 21-15
reboot, waiting until active sessions end 26-9
redirect, ICMP message B-9
redundancy, in site-to-site VPNs, using crypto maps 26-25
regular expression 8-10
reloading
  security appliance 33-8
RFCs for SNMP 30-18
router
  advertisement, ICMP message B-9
  solicitation, ICMP message B-9
routes
  about default 7-4
  configuring default routes 7-4
  configuring static routes 7-6
RSA
  keys, generating 17-3
RTSP inspection
  about 21-15
configuring 21-15
rules
   ICMP 17-8
running configuration
   copying 32-6
   saving 2-7

S
SAs, lifetimes 26-20
SCCP (Skinny) inspection
   about 21-25
   configuration 21-25
   configuring 21-25
SDI
   configuring 18-11
   support 18-6
secondary unit, failover 3-2
security appliance
   CLI  A-1
   connecting to 2-2
   managing the configuration 2-7
   reloading 33-8
   upgrading software 32-2
   viewing files in Flash memory 32-1
security association
   clearing 26-26
   See also SAs
security models for SNMP 30-6
sending messages to an e-mail address 29-9
sending messages to an SNMP server 29-11
sending messages to ASDM 29-10
sending messages to a specified output destination 29-15
sending messages to a syslog server 29-7
sending messages to a Telnet or SSH session 29-11
sending messages to the console port 29-10
sending messages to the internal log buffer 29-8
service policy
   applying 14-16
default 14-16
   interface 14-16
session management path 1-10
severity levels, of system log messages
   changing 29-1
   filtering by 29-1
   list of 29-3
severity levels, of system messages
   definition 29-3
SHA, IKE policy keywords (table) 26-4, 26-5
show command, filtering output  A-4
SIP inspection
   about 21-19
   configuring 21-19
   instant messaging 21-19
   timeouts 21-24
   troubleshooting 21-24
site-to-site VPNs, redundancy 26-25
Smart Call Home monitoring 31-19
SMTP inspection 20-25
SNMP
   about 30-1
   failover 30-8
   management station 29-1, 29-6
   prerequisites 30-7
SNMP configuration 30-8
SNMP groups 30-7
SNMP hosts 30-7
SNMP monitoring 30-15, 30-16
SNMP terminology 30-2
SNMP traps 30-3
SNMP users 30-7
SNMP Version 3 30-6, 30-12
SNMP Versions 1 and 2c 30-11
source quench, ICMP message B-9
speed, configuring 4-5
split tunneling
   group policy, domains 27-14
SSH
authentication 18-21
concurrent connections 17-2
login 17-4
password 5-2
RSA key 17-3
username 17-4
startup configuration
  copying 32-6
  saving 2-7
statd buffer overflow attack 25-8
Stateful Failover
  about 3-9
  state information 3-9
  state link 3-7
stateful inspection 1-10
  bypassing 24-3
state information 3-9
state link 3-7
static NAT
  about 11-4
  few-to-many mapping 11-8
  many-to-few mapping 11-7, 11-8
  network object NAT 12-9
twice NAT 13-15
static NAT with port translation
  about 11-4
static routes
  configuring 7-6
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 22-3
  configuring 22-3
syntax formatting A-3
syslogd server program 29-4
syslog messages
  analyzing 29-2
syslog messaging for SNMP 30-16
syslog server
  designating more than one as output destination 29-5
  EMBLEM format
    configuring 29-13
    enabling 29-7, 29-13
system log messages
  classes 29-3
  classes of 29-4
  configuring in groups
    by message list 29-4
    by severity level 29-1
device ID, including 29-16
disabling logging of
  filtering by message class 29-4
  managing in groups
    by message class 29-15
  output destinations 29-1, 29-6
  syslog message server 29-6
  Telnet or SSH session 29-6
severity levels
  about 29-3
  changing the severity level of a message 29-1
timestamp, including 29-16
T
TACACS+
  command authorization, configuring 18-31
  configuring a server 18-11
  support 18-6
TCP
  ports and literal values B-5
  sequence number randomization
disabling using Modular Policy Framework 24-12
TCP FIN only flags attack 25-7
TCP Intercept
  enabling using Modular Policy Framework 24-12
TCP normalization 24-3
TCP NULL flags attack 25-6
TCP state bypass
  AAA 24-5
  configuring 24-10
  inspection 24-5
  NAT 24-5
  SSMs and SSCs 24-5
  TCP Intercept 24-5
  TCP normalization 24-5
  unsupported features 24-5
TCP SYN+FIN flags attack 25-6
Telnet
  allowing management access 17-1
  authentication 18-21
  concurrent connections 17-2
  login 17-3
  password 5-2
testing configuration 33-1
time exceeded, ICMP message B-9
time ranges, access lists B-14
timezone, including in system log messages 29-16
timestamp reply, ICMP message B-9
timestamp request, ICMP message B-9
transform set
  definition 26-10
troubleshooting
  H.323 21-9
  H.323 RAS 21-10
  SIP 21-24
troubleshooting SNMP 30-13
tunnel
  IPsec 26-10
  security appliance as a tunnel endpoint 26-1
tunnel group
  configuring 27-3
  default 26-9, 27-1, 27-2
  default LAN-to-LAN, configuring 27-4
  definition 27-1, 27-2
  general parameters 27-2
  inheritance 27-1
  LAN-to-LAN, configuring 27-4
tunnel-group
  general attributes 27-2
tunneling, about 26-1
twice NAT
  about 11-15
  comparison with network object NAT 11-14
  configuring 13-1
dynamic NAT 13-3
dynamic PAT 13-8
elements 13-24
  guidelines 13-2
  identity NAT 13-20
  monitoring 13-24
  prerequisites 13-2
  static NAT 13-15
UDP
  bomb attack 25-7
  chargen DoS attack 25-7
  connection state information 1-10
  ports and literal values B-5
  snort attack 25-7
  unreachable, ICMP message B-9
  unreachable messages
    required for MTU discovery 17-8
  user, VPN
    definition 27-1
  user EXEC mode
    accessing 2-2
Index

prompt A-2
username
   adding 18-17
   encrypted 18-18
   password 18-18
users
   SNMP 30-7
U-turn 26-18

V

viewing RMS 32-9
virtual reassembly 1-9
VoIP
   proxy servers 21-19
   troubleshooting 21-9
VPN
   address range, subnets B-4
VPN attributes, group policy 27-10
VPN client
   NAT rules 11-18
VPN Client, IPsec attributes 26-2