CHAPTER 4

# 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.

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. 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.

**Note** 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 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, and an MIS group to access other parts. In addition, you might allow specific users within MIS to access systems that other MIS users cannot access. Connection profiles and group policies provide the flexibility to do so securely.

**Note** The ASA also includes the concept of object groups, which are a superset of network lists. Object groups let you define VPN access to ports as well as networks. Object groups relate to ACLs rather than to group policies and connection profiles. For more information about using object groups, see Chapter 20, "Configuring Objects," in the general operations configuration guide.

The security appliance can apply attribute values from a variety of sources. It applies them according to the following hierarchy:

1. Dynamic Access Policy (DAP) record
2. Username
3. Group policy
4. Group policy for the connection profile
5. Default group policy

Therefore, DAP values for an attribute have a higher priority than those configured for a user, group policy, or connection profile.

When you enable or disable an attribute for a DAP record, the ASA applies that value and enforces it. For example, when you disable HTTP proxy in dap webvpn mode, the ASA looks no further for a value. When you instead use the **no** value for the **http-proxy** command, the attribute is not present in the DAP record, so the security appliance moves down to the AAA attribute in the username, and if necessary, the group policy to find a value to apply. The ASA clientless SSL VPN configuration supports only one http-proxy and one https-proxy command each. We recommend that you use ASDM to configure DAP.

# 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 provides the following default connection profiles: DefaultL2Lgroup for LAN-to-LAN connections, DefaultRAgroup for remote access connections, and DefaultWEBVPNGroup for SSL VPN (browser-based) connections. 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 and are not configurable on external servers.

Connection profiles specify the following attributes:

- General Connection Profile Connection Parameters, page 4-3
- IPsec Tunnel-Group Connection Parameters, page 4-4
- Connection Profile Connection Parameters for SSL VPN Sessions, page 4-5

## 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. The following considerations apply:
  - For clients that use preshared keys to authenticate, the connection profile name is the same as the group name that a client passes to the ASA.
  - Clients that use certificates to authenticate pass this name as part of the certificate, and the ASA extracts the name from the certificate.
- Connection type—Connection types include IKEv1 remote-access, IPsec Lan-to-LAN, and Anyconnect (SSL/IKEv2). A connection profile can have only one connection type.
- Authentication, Authorization, and Accounting servers—These parameters identify the server groups or lists that the ASA 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 uses as defaults when authenticating or authorizing a tunnel user.
- Client address assignment method—This method includes values for one or more DHCP servers or address pools that the ASA assigns to clients.
- Override account disabled—This parameter lets you override the “account-disabled” indicator received from a AAA server.
- Password management—This parameter lets you warn a user that the current password is due to expire in a specified number of days (the default is 14 days), then offer the user the opportunity to change the password.
- Strip group and strip realm—These parameters direct the way the ASA processes the usernames it receives. They apply only to usernames received in the form user@realm.

  A realm is an administrative domain appended to a username with the @ delimiter (user@abc). If you strip the realm, the ASA uses the username and the group (if present) for authentication. If you strip the group, the ASA uses the username and the realm (if present) for authentication.

  Enter the strip-realm command to remove the realm qualifier, and enter the strip-group command to remove the group qualilfier from the username during authentication. If you remove both qualifiers, authentication is based on the *username* alone. Otherwise, authentication is based on the full *username@realm* or *username<delimiter> group* string. You must specify strip-realm if your server is unable to parse delimiters.

  In addition, for L2TP/IPsec clients only, when you specify the strip-group command the ASA selects the connection profile (tunnel group) for user connections by obtaining the group name from the username presented by the VPN client.
- Authorization required—This parameter lets you require authorization before a user can connect, or turn off that requirement.
- Authorization DN attributes—This parameter specifies which Distinguished Name attributes to use when performing authorization.

# IPsec Tunnel-Group Connection Parameters

IPsec parameters include the following:

- A client authentication method: preshared keys, certificates, or both.
  - For IKE connections based on preshared keys, this is the alphanumeric key itself (up to 128 characters long), associated with the connection policy.
  - Peer-ID validation requirement—This parameter specifies whether to require validating the identity of the peer using the peer's certificate.
  - If you specify certificates or both for the authentication method, the end user must provide a valid certificate in order to authenticate.
- An extended hybrid authentication method: XAUTH and hybrid XAUTH.

  You use **isakmp ikev1-user-authentication** command to implement hybrid XAUTH authentication when you need to use digital certificates for ASA authentication and a different, legacy method for remote VPN user authentication, such as RADIUS, TACACS+ or SecurID.
- ISAKMP (IKE) keepalive settings. This feature lets the ASA monitor the continued presence of a remote peer and report its own presence to that peer. If the peer becomes unresponsive, the ASA removes the connection. Enabling IKE keepalives prevents hung connections when the IKE peer loses connectivity.

  There are various forms of IKE keepalives. For this feature to work, both the ASA and its remote peer must support a common form. This feature works with the following peers:
  - Cisco AnyConnect VPN Client
  - Cisco VPN Client (Release 3.0 and above)
  - Cisco VPN 3000 Client (Release 2.x)
  - Cisco VPN 3002 Hardware Client
  - Cisco VPN 3000 Series Concentrators
  - Cisco IOS software
  - Cisco Secure PIX Firewall

  Non-Cisco VPN clients do not support IKE keepalives.

  If you are configuring a group of mixed peers, and some of those peers support IKE keepalives and others do not, enable IKE keepalives for the entire group. The feature does not affect the peers that do not support it.

  If you disable IKE keepalives, connections with unresponsive peers remain active until they time out, so we recommend that you keep your idle timeout short. To change your idle timeout, see "Configuring Group Policies" section on page 4-39.

**Note** To reduce connectivity costs, disable IKE keepalives if this group includes any clients connecting via ISDN lines. ISDN connections normally disconnect if idle, but the IKE keepalive mechanism prevents connections from idling and therefore from disconnecting.

If you do disable IKE keepalives, the client disconnects only when either its IKE or IPsec keys expire. Failed traffic does not disconnect the tunnel with the Peer Timeout Profile values as it does when IKE keepalives are enabled.

**Note** If you have a LAN-to-LAN configuration using IKE main mode, make sure that the two peers have the same IKE keepalive configuration. Both peers must have IKE keepalives enabled or both peers must have it disabled.

- If you configure authentication using digital certificates, you can specify whether to send the entire certificate chain (which sends the peer the identity certificate and all issuing certificates) or just the issuing certificates (including the root certificate and any subordinate CA certificates).
- You can notify users who are using outdated versions of Windows client software that they need to update their client, and you can provide a mechanism for them to get the updated client version. For VPN 3002 hardware client users, you can trigger an automatic update. You can configure and change the client-update, either for all connection profiles or for particular connection profiles.
- If you configure authentication using digital certificates, you can specify the name of the trustpoint that identifies the certificate to send to the IKE peer.

## Connection Profile Connection Parameters for SSL VPN Sessions

Table 4-1 provides a list of connection profile attributes that are specific to SSL VPN (AnyConnect client and clientless) connections. In addition to these attributes, you configure general connection profile attributes common to all VPN connections. For step-by-step information about configuring connection profiles, see Configuring Connection Profiles for Clientless SSL VPN Sessions, page 4-20.

**Note** In earlier releases, "connection profiles" were known as "tunnel groups." You configure a connection profile with tunnel-group commands. This chapter often uses these terms interchangeably.

*Table 4-1 Connection Profile Attributes for SSL VPN*

| Command | Function |
|---|---|
| **authentication** | Sets the authentication method, AAA or certificate. |
| **customization** | Identifies the name of a previously defined customization to apply. Customizations determine the appearance of the windows that the user sees upon login. You configure the customization parameters as part of configuring clientless SSL VPN. |
| **nbns-server** | Identifies the name of the NetBIOS Name Service server (nbns-server) to use for CIFS name resolution. |
| **group-alias** | Specifies one or more alternate names by which the server can refer to a connection profile. At login, the user selects the group name from a dropdown menu. |
| **group-url** | Identifies one or more group URLs. If you configure this attribute, users coming in on a specified URL need not select a group at login. |
| **dns-group** | Identifies the DNS server group that specifies the DNS server name, domain name, name server, number of retries, and timeout values for a DNS server to use for a connection profile. |
| **hic-fail-group-policy** | Specifies a VPN feature policy if you use the Cisco Secure Desktop Manager to set the Group-Based Policy attribute to "Use Failure Group-Policy" or "Use Success Group-Policy, if criteria match." |

*Table 4-1 Connection Profile Attributes for SSL VPN (continued)*

| Command | Function |
|---|---|
| **override-svc-download** | Overrides downloading the group-policy or username attributes configured for downloading the AnyConnect VPN client to the remote user. |
| **radius-reject-message** | Enables the display of the RADIUS reject message on the login screen when authentication is rejected. |

# Configuring Connection Profiles

This section describes the contents and configuration of connection profiles in both single context mode or multiple-context mode:

**Note** Multiple-context mode applies only to IKEv2 and IKEv1 site to site and does not apply to AnyConnect, Clientless SSL VPN, legacy Cisco VPN client, the Apple native VPN client, the Microsoft native VPN client, or cTCP for IKEv1 IPsec.

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 do not 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 can support is a function of the maximum number of concurrent VPN sessions for the platform + 5. For example, an ASA 5505 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 4-2 specifies the maximum VPN sessions and connection profiles for each ASA platform.

*Table 4-2 Maximum VPN Sessions and Connection Profiles Per ASA Platform*

| | 5505 Base/<br>Security Plus | 5510/Base/<br>Security Plus | 5520 | 5540 | 5550 | 5580-20 | 5580-40 |
|---|---|---|---|---|---|---|---|
| Maximum VPN Sessions | 10/25 | 250 | 750 | 5000 | 5000 | 10,000 | 10,000 |
| Maximum Connection Profiles | 15/30 | 255 | 755 | 5005 | 5005 | 10,005 | 10,005 |

# Default IPsec Remote Access Connection Profile Configuration

The contents of the default remote-access connection profile are as follows:

```
tunnel-group DefaultRAGroup type remote-access
tunnel-group DefaultRAGroup general-attributes
 no address-pool
 no ipv6-address-pool
 authentication-server-group LOCAL
 accounting-server-group RADIUS
 default-group-policy DfltGrpPolicy
 no dhcp-server
 no strip-realm
 no password-management
 no override-account-disable
 no strip-group
 no authorization-required
 authorization-dn-attributes CN OU
tunnel-group DefaultRAGroup webvpn-attributes
 hic-fail-group-policy DfltGrpPolicy
 customization DfltCustomization
 authentication aaa
 no override-svc-download
 no radius-reject-message
 dns-group DefaultDNS
tunnel-group DefaultRAGroup ipsec-attributes
 no pre-shared-key
 peer-id-validate req
 no chain
 no trust-point
 isakmp keepalive threshold 1500 retry 2
 no radius-sdi-xauth
 isakmp ikev1-user-authentication xauth
tunnel-group DefaultRAGroup ppp-attributes
 no authentication pap
 authentication chap
 authentication ms-chap-v1
 no authentication ms-chap-v2
 no authentication eap-proxy
```

# Configuring IPsec Tunnel-Group General Attributes

The general attributes are common across more than one tunnel-group type. IPsec remote access and clientless SSL VPN tunnels share most of the same general attributes. IPsec LAN-to-LAN tunnels use a subset. Refer to the *Cisco ASA Series Command Reference* for complete descriptions of all commands. This section describes, in order, how to configure remote-access and LAN-to-LAN connection profiles.

# Configuring Remote-Access Connection Profiles

Use a remote-access connection profile when setting up a connection between the following remote clients and a central-site ASA:

- Legacy Cisco VPN Client (connecting with IPsec/IKEv1)
- AnyConnect Secure Mobility Client (connecting with SSL or IPsec/IKEv2)
- Clientless SSL VPN (browser-based connecting with SSL)
- Cisco ASA 5500 Easy VPN hardware client (connecting with IPsec/IKEv1)
- Cisco VPM 3002 hardware client (connecting with IPsec/IKEv1)

We also provide a default group policy named *DfltGrpPolicy*.

To configure an remote-access connection profile, first configure the tunnel-group general attributes, then the remote-access attributes. See the following sections:

- Specifying a Name and Type for the Remote Access Connection Profile, page 4-8.
- Configuring Remote-Access Connection Profile General Attributes, page 4-8.
- Configuring Double Authentication, page 4-12
- Configuring Remote-Access Connection Profile IPsec IKEv1 Attributes, page 4-14.
- Configuring IPsec Remote-Access Connection Profile PPP Attributes, page 4-16

## Specifying a Name and Type for the Remote Access Connection Profile

Create the connection profile, specifying its name and type, by entering the **tunnel-group** command. For an remote-access tunnel, the type is **remote-access:**

```
hostname(config)# tunnel-group tunnel_group_name type remote-access
hostname(config)#
```

For example, to create an remote-access connection profile named TunnelGroup1, enter the following command:

```
hostname(config)# tunnel-group TunnelGroup1 type remote-access
hostname(config)#
```

## Configuring Remote-Access Connection Profile General Attributes

To configure or change the connection profile general attributes, specify the parameters in the following steps:

**Step 1** To configure the general attributes, enter the **tunnel-group general-attributes** task in either single or multiple context mode, which enters tunnel-group general-attributes configuration mode. The prompt changes to indicate the change in mode.

```
hostname(config)# tunnel-group tunnel_group_name general-attributes
hostname(config-tunnel-general)#
```

**Step 2** Specify the name of the authentication-server group, if any, to use. If you want to use the LOCAL database for authentication if the specified server group fails, append the keyword **LOCAL**:

```
hostname(config-tunnel-general)# authentication-server-group [(interface_name)] groupname
[LOCAL]
hostname(config-tunnel-general)#
```

The name of the authentication server group can be up to 16 characters long.

You can optionally configure interface-specific authentication by including the name of an interface after the group name. The interface name, which specifies where the tunnel terminates, must be enclosed in parentheses. The following command configures interface-specific authentication for the interface named test using the server named servergroup1 for authentication:

```
hostname(config-tunnel-general)# authentication-server-group (test) servergroup1
hostname(config-tunnel-general)#
```

**Step 3** Specify the name of the authorization-server group, if any, to use. When you configure this value, users must exist in the authorization database to connect:

```
hostname(config-tunnel-general)# authorization-server-group groupname
hostname(config-tunnel-general)#
```

The name of the authorization server group can be up to 16 characters long. For example, the following command specifies the use of the authorization-server group FinGroup:

```
hostname(config-tunnel-general)# authorization-server-group FinGroup
hostname(config-tunnel-general)#
```

**Step 4** Specify the name of the accounting-server group, if any, to use:

```
hostname(config-tunnel-general)# accounting-server-group groupname
hostname(config-tunnel-general)#
```

The name of the accounting server group can be up to 16 characters long. For example, the following command specifies the use of the accounting-server group named comptroller:

```
hostname(config-tunnel-general)# accounting-server-group comptroller
hostname(config-tunnel-general)#
```

**Step 5** Specify the name of the default group policy:

```
hostname(config-tunnel-general)# default-group-policy policyname
hostname(config-tunnel-general)#
```

The name of the group policy can be up to 64 characters long. The following example sets DfltGrpPolicy as the name of the default group policy:

```
ciscoasa(config-tunnel-general)# default-group-policy DfltGrpPolicy
hostname(config-tunnel-general)#
```

**Step 6** Specify the names or IP addresses of the DHCP server (up to 10 servers), and the names of the DHCP address pools (up to 6 pools). The defaults are no DHCP server and no address pool. The **dhcp-server** command will allow you to configure the ASA to send additional options to the specified DHCP servers when it is trying to get IP addresses for VPN clients. See the **dhcp-server** command in the *Cisco ASA Series Command Reference* guide for more information.

```
hostname(config-tunnel-general)# dhcp-server server1 [...server10]
hostname(config-tunnel-general)# address-pool [(interface name)] address_pool1
[...address_pool6]
hostname(config-tunnel-general)#
```

**Note** If you specify an interface name, you must enclosed it within parentheses.

You configure address pools with the **ip local pool** command in global configuration mode.

**Step 7** Specify the name of the NAC authentication server group, if you are using Network Admission Control, to identify the group of authentication servers to be used for Network Admission Control posture validation. Configure at least one Access Control Server to support NAC. Use the **aaa-server** command to name the ACS group. Then use the **nac-authentication-server-group** command, using the same name for the server group.

The following example identifies acs-group1 as the authentication server group to be used for NAC posture validation:

```
ciscoasa(config-group-policy)# nac-authentication-server-group acs-group1
ciscoasa(config-group-policy)
```

The following example inherits the authentication server group from the default remote access group:

```
ciscoasa(config-group-policy)# no nac-authentication-server-group
ciscoasa(config-group-policy)
```

**Note** NAC requires a Cisco Trust Agent on the remote host.

**Step 8** Specify whether to strip the group or the realm from the username before passing it on to the AAA server. The default is not to strip either the group name or the realm:

```
hostname(config-tunnel-general)# strip-group
hostname(config-tunnel-general)# strip-realm
hostname(config-tunnel-general)#
```

A realm is an administrative domain. If you strip the realm, the ASA uses the username and the group (if present) authentication. If you strip the group, the ASA uses the username and the realm (if present) for authentication. Enter the **strip-realm** command to remove the realm qualifier, and use the strip-group command to remove the group qualilfier from the username during authentication. If you remove both qualifiers, authentication is based on the *username* alone. Otherwise, authentication is based on the full *username@realm* or *username<delimiter> group* string. You must specify **strip-realm** if your server is unable to parse delimiters.

**Step 9** Optionally, if your server is a RADIUS, RADIUS with NT, or LDAP server, you can enable password management.

**Note** If you are using an LDAP directory server for authentication, password management is supported with the Sun Microsystems JAVA System Directory Server (formerly named the Sun ONE Directory Server) and the Microsoft Active Directory.

Sun—The DN configured on the ASA 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 ACI on the default password policy.

Microsoft—You must configure LDAP over SSL to enable password management with Microsoft Active Directory.

See the general operations configuration guide for more information.

This feature, which is disabled by default, warns a user when the current password is about to expire. The default is to begin warning the user 14 days before expiration:

```
hostname(config-tunnel-general)# password-management
hostname(config-tunnel-general)#
```

If the server is an LDAP server, you can specify the number of days (0 through 180) before expiration to begin warning the user about the pending expiration:

```
hostname(config-tunnel-general)# password-management [password-expire in days n]
hostname(config-tunnel-general)#
```

**Note** The **password-management** command, entered in tunnel-group general-attributes configuration mode replaces the deprecated **radius-with-expiry** command that was formerly entered in tunnel-group ipsec-attributes mode.

When you configure the **password-management** command, the ASA notifies the remote user at login that the user's current password is about to expire or has expired. The ASA then offers the user the opportunity to change the password. If the current password has not yet expired, the user can still log in using that password. The ASA ignores this command if RADIUS or LDAP authentication has not been configured.

Note that this does not change the number of days before the password expires, but rather, the number of days ahead of expiration that the ASA starts warning the user that the password is about to expire.

If you do specify the **password-expire-in-days** keyword, you must also specify the number of days.

Specifying this command with the number of days set to 0 disables this command. The ASA does not notify the user of the pending expiration, but the user can change the password after it expires.

See Configuring Microsoft Active Directory Settings for Password Management, page 4-28 for more information.

**Note** The ASA Version 7.1 and later generally supports password management for the AnyConnect VPN Client, the Cisco IPsec VPN Client, the SSL VPN full-tunneling client, and Clientless connections when authenticating with LDAP or with any RADIUS connection that supports MS-CHAPv2. Password management is *not* supported for any of these connection types for Kerberos/AD (Windows password) or NT 4.0 Domain.

Some RADIUS servers that support MS-CHAP do not currently support MS-CHAPv2. The **password-management** command requires MS-CHAPv2, so please check with your vendor.

The RADIUS server (for example, Cisco ACS) could proxy the authentication request to another authentication server. However, from the ASA perspective, it is talking only to a RADIUS server.

For LDAP, the method to change a password is proprietary for the different LDAP servers on the market. Currently, the ASA implements the proprietary password management logic only for Microsoft Active Directory and Sun LDAP servers. Native LDAP requires an SSL connection. You must enable LDAP over SSL before attempting to do password management for LDAP. By default, LDAP uses port 636.

**Step 10** Optionally, configure the ability to override an account-disabled indicator from a AAA server, by entering the **override-account-disable** command:

```
hostname(config-tunnel-general)# override-account-disable
hostname(config-tunnel-general)#
```

**Note** Allowing override-account-disable is a potential security risk.

**Step 11** Specify the attribute or attributes to use in deriving a name for an authorization query from a certificate. This attribute specifies what part of the subject DN field to use as the username for authorization:

```
ciscoasa(config-tunnel-general)# authorization-dn-attributes {primary-attribute
[secondary-attribute] | use-entire-name}
```

For example, the following command specifies the use of the CN attribute as the username for authorization:

```
ciscoasa(config-tunnel-general)# authorization-dn-attributes CN
ciscoasa(config-tunnel-general)#
```

The authorization-dn-attributes are **C** (Country), **CN** (Common Name), **DNQ** (DN qualifier), **EA** (E-mail Address), **GENQ** (Generational qualifier), **GN** (Given Name), **I** (Initials), **L** (Locality), **N** (Name), **O** (Organization), **OU** (Organizational Unit), **SER** (Serial Number), **SN** (Surname), **SP** (State/Province), **T** (Title), **UID** (User ID), and **UPN** (User Principal Name).

**Step 12** Specify whether to require a successful authorization before allowing a user to connect. The default is not to require authorization.

```
ciscoasa(config-tunnel-general)# authorization-required
ciscoasa(config-tunnel-general)#
```

## Configuring Double Authentication

Double authentication is an optional feature that requires a user to enter an additional authentication credential, such as a second username and password, on the login screen. Specify the following commands to configure double authentication.

**Step 1** Specify the secondary authentication server group. This command specifies the AAA server group to use as the secondary AAA server.

**Note** This command applies only to AnyConnect client VPN connections.

The secondary server group cannot specify an SDI server group. By default, no secondary authentication is required.

```
ciscoasa(config-tunnel-general)# secondary-authentication-server-group [interface_name]
{none | LOCAL | groupname [LOCAL]} [use-primary-name]
```

If you use the none keyword, no secondary authentication is required. The *groupname* value specifies the AAA server group name. Local specifies the use of the internal server database, and when used with the groupname value, LOCAL specifies fallback. For example, to set the primary authentication server group to sdi_group and the secondary authentication server group to ldap_server, enter the following commands:

```
ciscoasa(config-tunnel-general)# authentication-server-group
ciscoasa(config-tunnel-general)# secondary-authentication-server-group
```

**Note** If you use the **use-primary-name** keyword, then the login dialog requests only one username. In addition, if the usernames are extracted from a digital certificate, only the primary username is used for authentication.

**Step 2** If obtaining the secondary username from a certificate, enter **secondary-username-from-certificate**:

```
ciscoasa(config-tunnel-general)# secondary-username-from-certificate C | CN | ... |
use-script
```

The values for the DN fields to extract from the certificate for use as a secondary username are the same as for the primary **username-from-certificate** command. Alternatively, you can specify the use-script keyword, which directs the ASA to use a script file generated by ASDM.

For example, to specify the Common Name as the primary username field and Organizational Unit as the secondary username field, enter the following commands:

```
ciscoasa(config-tunnel-general)# tunnel-group test1 general-attributes
ciscoasa(config-tunnel-general)# username-from-certificate cn
ciscoasa(config-tunnel-general)# secondary-username-from-certificate ou
```

**Step 3** Use the **secondary-pre-fill-username** command in tunnel-group webvpn-attributes mode to enable extracting a secondary username from a client certificate for use in authentication. Use the keywords to specify whether this command applies to a clientless connection or an SSL VPN (AnyConnect) client connection and whether you want to hide the extracted username from the end user. This feature is disabled by default. Clientless and SSL-client options can both exist at the same time, but you must configure them in separate commands.

```
ciscoasa(config-tunnel-general)# secondary-pre-fill-username-from-certificate {clientless
| ssl-client} [hide]
```

For example, to specify the use of pre-fill-username for both the primary and secondary authentication for a connection, enter the following commands:

```
ciscoasa(config-tunnel-general)# tunnel-group test1 general-attributes
ciscoasa(config-tunnel-general)# pre-fill-username ssl-client
ciscoasa(config-tunnel-general)# secondary-pre-fill-username ssl-client
```

**Step 4** Specify which authentication server to use to obtain the authorization attributes to apply to the connection. The primary authentication server is the default selection. This command is meaningful only for double authentication.

```
ciscoasa(config-tunnel-general)# authentication-attr-from-server {primary | secondary}
```

For example, to specify the use of the secondary authentication server, enter the following commands:

```
ciscoasa(config-tunnel-general)# tunnel-group test1 general-attributes
ciscoasa(config-tunnel-general)# authentication-attr-from-server secondary
```

**Step 5** Specify which authentication username, primary or secondary, to associate with the session. The default value is primary. With double authentication enabled, it is possible that two distinct usernames are authenticated for the session. The administrator must designate one of the authenticated usernames as the session username. The session username is the username provided for accounting, session database, syslogs, and debug output.

```
ciscoasa(config-tunnel-general)# authenticated-session-username {primary | secondary}
```

For example, to specify that the authentication username associated with the session must come from the secondary authentication server, enter the following commands:

```
ciscoasa(config-tunnel-general)# tunnel-group test1 general-attributes
ciscoasa(config-tunnel-general)# authenticated-session-username secondary
```

## Configuring Remote-Access Connection Profile IPsec IKEv1 Attributes

To configure the IPsec IKEv1 attributes for a remote-access connection profile, perform the following steps. The following description assumes that you have already created the remote-access connection profile. Remote-access connection profiles have more attributes than LAN-to-LAN connection profiles.

**Step 1** To specify the IPsec attributes of an remote-access tunnel-group, enter tunnel-group ipsec-attributes mode by entering the following command in either single or multiple context mode. The prompt changes to indicate the mode change.

```
ciscoasa(config)# tunnel-group tunnel-group-name ipsec-attributes
ciscoasa(config-tunnel-ipsec)#
```

This command enters tunnel-group ipsec-attributes configuration mode, in which you configure the remote-access tunnel-group IPsec attributes in either single or multiple context mode.

For example, the following command designates that the tunnel-group ipsec-attributes mode commands that follow pertain to the connection profile named TG1. Notice that the prompt changes to indicate that you are now in tunnel-group ipsec-attributes mode:

```
ciscoasa(config)# tunnel-group TG1 type remote-access
ciscoasa(config)# tunnel-group TG1 ipsec-attributes
ciscoasa(config-tunnel-ipsec)#
```

**Step 2** Specify the preshared key to support IKEv1 connections based on preshared keys. For example, the following command specifies the preshared key xyzx to support IKEv1 connections for an IPsec IKEv1 remote access connection profile:

```
ciscoasa(config-tunnel-ipsec)# ikev1 pre-shared-key xyzx
ciscoasa(config-tunnel-ipsec)#
```

**Step 3** Specify whether to validate the identity of the peer using the peer's certificate:

```
ciscoasa(config-tunnel-ipsec)# peer-id-validate option
ciscoasa(config-tunnel-ipsec)#
```

The possible *option* values are **req** (required), **cert** (if supported by certificate), and **nocheck** (do not check). The default is **req**.

For example, the following command specifies that peer-id validation is required:

```
ciscoasa(config-tunnel-ipsec)# peer-id-validate req
ciscoasa(config-tunnel-ipsec)#
```

**Step 4** Specify whether to enable sending of a certificate chain. The following command includes the root certificate and any subordinate CA certificates in the transmission:

```
ciscoasa(config-tunnel-ipsec)# chain
ciscoasa(config-tunnel-ipsec)#
```

This attribute applies to all IPsec tunnel-group types.

**Step 5** Specify the name of a trustpoint that identifies the certificate to be sent to the IKE peer:

```
ciscoasa(config-tunnel-ipsec)# ikev1 trust-point trust-point-name
ciscoasa(config-tunnel-ipsec)#
```

The following command specifies mytrustpoint as the name of the certificate to be sent to the IKE peer:

```
ciscoasa(config-ipsec)# ikev1 trust-point mytrustpoint
```

**Step 6** Specify the ISAKMP keepalive threshold and the number of retries allowed:

```
ciscoasa(config-tunnel-ipsec)# isakmp keepalive threshold <number> retry <number>
ciscoasa(config-tunnel-ipsec)#
```

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 ISAKMP keepalives, enter **isakmp keepalive disable**.

For example, the following command sets the IKE keepalive threshold value to 15 seconds and sets the retry interval to 10 seconds:

```
ciscoasa(config-tunnel-ipsec)# isakmp keepalive threshold 15 retry 10
ciscoasa(config-tunnel-ipsec)#
```

The default value for the **threshold** parameter is 300 for remote-access and 10 for LAN-to-LAN, and the default value for the retry parameter is 2.

To specify that the central site (secure gateway) should never initiate ISAKMP monitoring, enter the following command:

```
ciscoasa(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 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 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.

**Note** Before the authentication type can be set to hybrid, you must configure the authentication server, create a preshared key, and configure a trustpoint.

You can use the **isakmp ikev1-user-authentication** command with the optional **interface** parameter to specify a particular interface. When you omit the **interface** parameter, the command applies to all the interfaces and serves as a back-up when the per-interface command is not specified. When there are two **isakmp ikev1-user-authentication** commands specified for a connection profile, and one uses the **interface** parameter and one does not, the one specifying the interface takes precedence for that particular interface.

For example, the following commands enable hybrid XAUTH on the inside interface for a connection profile called example-group:

```
ciscoasa(config)# tunnel-group example-group type remote-access
ciscoasa(config)# tunnel-group example-group ipsec-attributes
ciscoasa(config-tunnel-ipsec)# isakmp ikev1-user-authentication (inside) hybrid
ciscoasa(config-tunnel-ipsec)#
```

## Configuring IPsec Remote-Access Connection Profile PPP Attributes

To configure the Point-to-Point Protocol attributes for a remote-access connection profile, perform the following steps. PPP attributes apply *only* to IPsec remote-access connection profiles. The following description assumes that you have already created the IPsec remote-access connection profile.

**Step 1** Enter tunnel-group ppp-attributes configuration mode, in which you configure the remote-access tunnel-group PPP attributes, by entering the following command. The prompt changes to indicate the mode change:

```
ciscoasa(config)# tunnel-group tunnel-group-name type remote-access
ciscoasa(config)# tunnel-group tunnel-group-name ppp-attributes
ciscoasa(config-tunnel-ppp)#
```

For example, the following command designates that the tunnel-group ppp-attributes mode commands that follow pertain to the connection profile named TG1. Notice that the prompt changes to indicate that you are now in tunnel-group ppp-attributes mode:

```
ciscoasa(config)# tunnel-group TG1 type remote-access
ciscoasa(config)# tunnel-group TG1 ppp-attributes
ciscoasa(config-tunnel-ppp)#
```

**Step 2** Specify whether to enable authentication using specific protocols for the PPP connection. The protocol value can be any of the following:

- pap—Enables the use of Password Authentication Protocol for the PPP connection.
- chap—Enables the use of Challenge Handshake Authentication Protocol for the PPP connection.
- ms-chap-v1 or ms-chap-v2—Enables the use of Microsoft Challenge Handshake Authentication Protocol, version 1 or version 2 for the PPP connection.
- eap—Enables the use of Extensible Authentication protocol for the PPP connection.

CHAP and MSCHAPv1 are enabled by default.

The syntax of this command is:

```
ciscoasa(config-tunnel-ppp)# authentication protocol
ciscoasa(config-tunnel-ppp)#
```

To disable authentication for a specific protocol, use the **no** form of the command:

```
ciscoasa(config-tunnel-ppp)# no authentication protocol
ciscoasa(config-tunnel-ppp)#
```

For example, the following command enables the use of the PAP protocol for a PPP connection:

```
ciscoasa(config-tunnel-ppp)# authentication pap
ciscoasa(config-tunnel-ppp)#
```

The following command enables the use of the MS-CHAP, version 2 protocol for a PPP connection:

```
ciscoasa(config-tunnel-ppp)# authentication ms-chap-v2
ciscoasa(config-tunnel-ppp)#
```

The following command enables the use of the EAP-PROXY protocol for a PPP connection:

```
ciscoasa(config-tunnel-ppp)# authentication pap
ciscoasa(config-tunnel-ppp)#
```

The following command disables the use of the MS-CHAP, version 1 protocol for a PPP connection:

```
ciscoasa(config-tunnel-ppp)# no authentication ms-chap-v1
ciscoasa(config-tunnel-ppp)#
```

## 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 4-17
- Configuring LAN-to-LAN Connection Profile General Attributes, page 4-17
- Configuring LAN-to-LAN IPsec IKEv1 Attributes, page 4-18

### 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
hostname(config)#
```

### Configuring LAN-to-LAN Connection Profile General Attributes

To configure the connection profile general attributes, perform the following steps:

**Step 1** Enter tunnel-group general-attributes mode by specifying the general-attributes keyword in either single or multiple context mode:

```
hostname(config)# tunnel-group_tunnel-group-name general-attributes
hostname(config-tunnel-general)#
```

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:

```
ciscoasa(config-tunnel-general)# default-group-policy MyPolicy
hostname(config-tunnel-general)#
```

## Configuring LAN-to-LAN IPsec IKEv1 Attributes

To configure the IPsec IKEv1 attributes, perform 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 in either single or multiple context mode.

```
ciscoasa(config)# tunnel-group tunnel-group-name ipsec-attributes
hostname(config-tunnel-ipsec)#
```

For example, the following command enters config-ipsec mode so that you can configure the parameters for the connection profile named TG1:

```
ciscoasa(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.

```
ciscoasa(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:

```
ciscoasa(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:

```
ciscoasa(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**:

```
ciscoasa(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:

```
ciscoasa(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:

```
ciscoasa(config-tunnel-ipsec)# trust-point trust-point-name
hostname(config-tunnel-ipsec)#
```

For example, the following command sets the trustpoint name to mytrustpoint:

```
ciscoasa(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:

```
ciscoasa(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 (secure gateway) should never initiate ISAKMP monitoring, enter the following command:

```
ciscoasa(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 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 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.

**Note** 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:

```
ciscoasa(config)# tunnel-group example-group type remote-access
ciscoasa(config)# tunnel-group example-group ipsec-attributes
ciscoasa(config-tunnel-ipsec)# isakmp ikev1-user-authentication hybrid
ciscoasa(config-tunnel-ipsec)#
```

# Configuring Connection Profiles for Clientless SSL VPN Sessions

The tunnel-group general attributes for clientless SSL VPN connection profiles are the same as those for IPsec remote-access connection profiles, except that the tunnel-group type is webvpn and the **strip-group** and **strip-realm** commands do not apply. You define the attribute specific to clientless SSL VPN separately. The following sections describe how to configure clientless SSL VPN connection profiles:

- Configuring General Tunnel-Group Attributes for Clientless SSL VPN Sessions, page 4-20
- Configuring Tunnel-Group Attributes for Clientless SSL VPN Sessions, page 4-23

## Configuring General Tunnel-Group Attributes for Clientless SSL VPN Sessions

To configure or change the connection profile general attributes, specify the parameters in the following steps.

**Step 1** To configure the general attributes, enter **tunnel-group general-attributes** command, which enters tunnel-group general-attributes configuration mode in either single or multiple context mode. Note that the prompt changes:

```
hostname(config)# tunnel-group tunnel_group_name general-attributes
hostname(config-tunnel-general)#
```

To configure the general attributes for TunnelGroup3, created in the previous section, enter the following command:

```
hostname(config)# tunnel-group TunnelGroup3 general-attributes
hostname(config-tunnel-general)#
```

**Step 2** Specify the name of the authentication-server group, if any, to use. If you want to use the LOCAL database for authentication if the specified server group fails, append the keyword LOCAL:

```
hostname(config-tunnel-general)# authentication-server-group groupname [LOCAL]
hostname(config-tunnel-general)#
```

For example, to configure the authentication server group named test, and to provide fallback to the LOCAL server if the authentication server group fails, enter the following command:

```
hostname(config-tunnel-general)# authentication-server-group test LOCAL
hostname(config-tunnel-general)#
```

The authentication-server-group name identifies a previously configured authentication server or group of servers. Use the **aaa-server** command to configure authentication servers. The maximum length of the group tag is 16 characters.

You can also configure interface-specific authentication by including the name of an interface in parentheses before the group name. The following interfaces are available by default:

- inside—Name of interface GigabitEthernet0/1
- outside— Name of interface GigabitEthernet0/0

**Note** The ASA's outside interface address (for both IPv4/IPv6) cannot overlap with the private side address space.

Other interfaces you have configured (using the **interface** command) are also available. The following command configures interface-specific authentication for the interface named outside using the server servergroup1 for authentication:

```
hostname(config-tunnel-general)# authentication-server-group (outside) servergroup1
hostname(config-tunnel-general)#
```

**Step 3** Optionally, specify the name of the authorization-server group, if any, to use. If you are not using authorization, go to Step 6. When you configure this value, users must exist in the authorization database to connect:

```
hostname(config-tunnel-general)# authorization-server-group groupname
hostname(config-tunnel-general)#
```

Use the **aaa-server** command to configure authorization servers. The maximum length of the group tag is 16 characters.

For example, the following command specifies the use of the authorization-server group FinGroup:

```
hostname(config-tunnel-general)# authorization-server-group FinGroup
hostname(config-tunnel-general)#
```

**Step 4** Specify whether to require a successful authorization before allowing a user to connect. The default is not to require authorization.

```
ciscoasa(config-tunnel-general)# authorization-required
ciscoasa(config-tunnel-general)#
```

**Step 5** Specify the attribute or attributes to use in deriving a name for an authorization query from a certificate. This attribute specifies what part of the subject DN field to use as the username for authorization:

```
ciscoasa(config-tunnel-general)# authorization-dn-attributes {primary-attribute
[secondary-attribute] | use-entire-name}
```

For example, the following command specifies the use of the CN attribute as the username for authorization:

```
ciscoasa(config-tunnel-general)# authorization-dn-attributes CN
ciscoasa(config-tunnel-general)#
```

The authorization-dn-attributes are **C** (Country), **CN** (Common Name), **DNQ** (DN qualifier), **EA** (E-mail Address), **GENQ** (Generational qualifier), **GN** (Given Name), **I** (Initials), **L** (Locality), **N** (Name), **O** (Organization), **OU** (Organizational Unit), **SER** (Serial Number), **SN** (Surname), **SP** (State/Province), **T** (Title), **UID** (User ID), and **UPN** (User Principal Name).

**Step 6** Optionally, specify the name of the accounting-server group, if any, to use. If you are not using accounting, go to Step 7. Use the **aaa-server** command to configure accounting servers. The maximum length of the group tag is 16 characters.:

```
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 comptroller:

```
hostname(config-tunnel-general)# accounting-server-group comptroller
hostname(config-tunnel-general)#
```

**Step 7** Optionally, specify the name of the default group policy. The default value is DfltGrpPolicy:

```
hostname(config-tunnel-general)# default-group-policy policyname
hostname(config-tunnel-general)#
```

The following example sets MyDfltGrpPolicy as the name of the default group policy:

```
ciscoasa(config-tunnel-general)# default-group-policy MyDfltGrpPolicy
hostname(config-tunnel-general)#
```

**Step 8** Optionally, specify the name or IP address of the DHCP server (up to 10 servers), and the names of the DHCP address pools (up to 6 pools). Separate the list items with spaces. The defaults are no DHCP server and no address pool.

```
hostname(config-tunnel-general)# dhcp-server server1 [...server10]
hostname(config-tunnel-general)# address-pool [(interface name)] address_pool1
[...address_pool6]
hostname(config-tunnel-general)#
```

**Note** The interface name must be enclosed in parentheses.

You configure address pools with the **ip local pool** command in global configuration mode. See Chapter 5, "Configuring IP Addresses for VPNs" for information about configuring address pools.

**Step 9** Optionally, if your server is a RADIUS, RADIUS with NT, or LDAP server, you can enable password management.

**Note** If you are using an LDAP directory server for authentication, password management is supported with the Sun Microsystems JAVA System Directory Server (formerly named the Sun ONE Directory Server) and the Microsoft Active Directory.

- Sun—The DN configured on the ASA 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 ACI on the default password policy.
- Microsoft—You must configure LDAP over SSL to enable password management with Microsoft Active Directory.

See the general operations configuration guide for more information.

This feature, which is enabled by default, warns a user when the current password is about to expire. The default is to begin warning the user 14 days before expiration:

```
hostname(config-tunnel-general)# password-management
hostname(config-tunnel-general)#
```

If the server is an LDAP server, you can specify the number of days (0 through 180) before expiration to begin warning the user about the pending expiration:

```
hostname(config-tunnel-general)# password-management [password-expire in days n]
hostname(config-tunnel-general)#
```

**Note** The **password-management** command, entered in tunnel-group general-attributes configuration mode replaces the deprecated **radius-with-expiry** command that was formerly entered in tunnel-group ipsec-attributes mode.

When you configure this command, the ASA notifies the remote user at login that the user's current password is about to expire or has expired. The ASA then offers the user the opportunity to change the password. If the current password has not yet expired, the user can still log in using that password. The ASA ignores this command if RADIUS or LDAP authentication has not been configured.

Note that this does not change the number of days before the password expires, but rather, the number of days ahead of expiration that the ASA starts warning the user that the password is about to expire.

If you do specify the **password-expire-in-days** keyword, you must also specify the number of days.

See Configuring Microsoft Active Directory Settings for Password Management, page 4-28 for more information.

**Step 10** Specifying this command with the number of days set to 0 disables this command. The ASA does not notify the user of the pending expiration, but the user can change the password after it expires.Optionally, configure the ability to override an account-disabled indicator from the AAA server, by entering the **override-account-disable** command:

```
hostname(config-tunnel-general)# override-account-disable
hostname(config-tunnel-general)#
```

**Note** Allowing override account-disabled is a potential security risk.

## Configuring Tunnel-Group Attributes for Clientless SSL VPN Sessions

To configure the parameters specific to a clientless SSL VPN connection profile, follow the steps in this section. Clientless SSL VPN was formerly known as WebVPN, and you configure these attributes in tunnel-group webvpn-attributes mode.

**Step 1** To specify the attributes of a clientless SSL VPN tunnel-group, enter tunnel-group webvpn-attributes mode by entering the following command. The prompt changes to indicate the mode change:

```
ciscoasa(config)# tunnel-group tunnel-group-name webvpn-attributes
ciscoasa(config-tunnel-ipsec)#
```

For example, to specify the webvpn-attributes for the clientless SSL VPN tunnel-group named sales, enter the following command:

```
ciscoasa(config)# tunnel-group sales webvpn-attributes
ciscoasa(config-tunnel-webvpn)#
```

**Step 2** To specify the authentication method to use: AAA, digital certificates, or both, enter the **authentication** command. You can specify either aaa or certificate or both, in any order.

```
ciscoasa(config-tunnel-webvpn)# authentication authentication_method
ciscoasa(config-tunnel-webvpn)#
```

For example, The following command allows both AAA and certificate authentication:

```
ciscoasa(config-tunnel-webvpn)# authentication aaa certificate
ciscoasa(config-tunnel-webvpn)#
```

### Applying Customization

Customizations determine the appearance of the windows that the user sees upon login. You configure the customization parameters as part of configuring clientless SSL VPN.

To apply a previously defined web-page customization to change the look-and-feel of the web page that the user sees at login, enter the customization command in username webvpn configuration mode:

```
ciscoasa(config-username-webvpn)# customization {none | value customization_name}
ciscoasa(config-username-webvpn)#
```

For example, to use the customization named blueborder, enter the following command:

```
ciscoasa(config-username-webvpn)# customization value blueborder
ciscoasa(config-username-webvpn)#
```

You configure the customization itself by entering the **customization** command in webvpn mode.

The following example shows a command sequence that first establishes a customization named “123” that defines a password prompt. The example then defines a clientless SSL VPN tunnel-group named “test” and uses the **customization** command to specify the use of the customization named “123”:

```
ciscoasa(config)# webvpn
ciscoasa(config-webvpn)# customization 123
ciscoasa(config-webvpn-custom)# password-prompt Enter password
ciscoasa(config-webvpn)# exit
ciscoasa(config)# tunnel-group test type webvpn
ciscoasa(config)# tunnel-group test webvpn-attributes
ciscoasa(config-tunnel-webvpn)# customization value 123
ciscoasa(config-tunnel-webvpn)#
```

**Step 3** The ASA queries NetBIOS name servers to map NetBIOS names to IP addresses. Clientless SSL VPN requires NetBIOS to access or share files on remote systems. Clientless SSL VPN uses NetBIOS and the CIFS protocol to access or share files on remote systems. When you attempt a file-sharing connection to a Windows computer by using its computer name, the file server you specify corresponds to a specific NetBIOS name that identifies a resource on the network.

To make the NBNS function operational, you must configure at least one NetBIOS server (host). You can configure up to three NBNS servers for redundancy. The ASA uses the first server on the list for NetBIOS/CIFS name resolution. If the query fails, it uses the next server.

To specify the name of the NBNS (NetBIOS Name Service) server to use for CIFS name resolution, use the **nbns-server** command. You can enter up to three server entries. The first server you configure is the primary server, and the others are backups, for redundancy. You can also specify whether this is a master browser (rather than just a WINS server), the timeout interval, and the number of retries. A WINS server or a master browser is typically on the same network as the ASA, or reachable from that network. You must specify the timeout interval before the number of retries:

```
ciscoasa(config-tunnel-webvpn)# nbns-server {host-name | IP_address} [master]
[timeout seconds] [retry number]
ciscoasa(config-tunnel-webvpn)#
```

For example, to configure the server named nbnsprimary as the primary server and the server 192.168.2.2 as the secondary server, each allowing three retries and having a 5-second timeout, enter the following command:

```
ciscoasa(config)# name 192.168.2.1 nbnsprimary
ciscoasa(config-tunnel-webvpn)# nbns-server nbnsprimary master timeout 5 retry 3
ciscoasa(config-tunnel-webvpn)# nbns-server 192.168.2.2 timeout 5 retry 3
ciscoasa(config-tunnel-webvpn)#
```

The timeout interval can range from 1 through 30 seconds (default 2), and the number of retries can be in the range 0 through 10 (default 2).

The **nbns-server** command in tunnel-group webvpn-attributes configuration mode replaces the deprecated **nbns-server** command in webvpn configuration mode.

**Step 4** To specify alternative names for the group, use the **group-alias** command. Specifying the group alias creates one or more alternate names by which the user can refer to a tunnel-group. The group alias that you specify here appears in the drop-down list on the user's login page. Each group can have multiple aliases or no alias, each specified in separate commands. This feature is useful when the same group is known by several common names, such as "Devtest" and "QA".

For each group alias, enter a **group-alias** command. Each alias is enabled by default. You can optionally explicitly enable or disable each alias:

```
ciscoasa(config-tunnel-webvpn)# group-alias alias [enable | disable]
ciscoasa(config-tunnel-webvpn)#
```

For example, to enable the aliases QA and Devtest for a tunnel-group named QA, enter the following commands:

```
ciscoasa(config-tunnel-webvpn)# group-alias QA enable
ciscoasa(config-tunnel-webvpn)# group-alias Devtest enable
ciscoasa(config-tunnel-webvpn)#
```

**Note** The webvpn tunnel-group-list must be enabled for the (dropdown) group list to appear.

**Step 5** To specify incoming URLs or IP addresses for the group, use the **group-url** command. Specifying a group URL or IP address eliminates the need for the user to select a group at login. When a user logs in, the ASA looks for the user's incoming URL or address in the tunnel-group-policy table. If it finds the URL or address and if group-url is enabled in the connection profile, then the ASA automatically selects the associated connection profile and presents the user with only the username and password fields in the login window. This simplifies the user interface and has the added advantage of never exposing the list of groups to the user. The login window that the user sees uses the customizations configured for that connection profile.

If the URL or address is disabled and group-alias is configured, then the dropdown list of groups is also displayed, and the user must make a selection.

You can configure multiple URLs or addresses (or none) for a group. Each URL or address can be enabled or disabled individually. You must use a separate **group-url** command for each URL or address specified. You must specify the entire URL or address, including either the http or https protocol.

You cannot associate the same URL or address with multiple groups. The ASA verifies the uniqueness of the URL or address before accepting the URL or address for a connection profile.

For each group URL or address, enter a **group-url** command. You can optionally explicitly enable (the default) or disable each URL or alias:

```
ciscoasa(config-tunnel-webvpn)# group-url url [enable | disable]
ciscoasa(config-tunnel-webvpn)#
```

*Url* specifies a URL or IP address for this tunnel group.

For example, to enable the group URLs http://www.example.com and http://192.168.10.10 for the tunnel-group named RadiusServer, enter the following commands:

```
ciscoasa(config)# tunnel-group RadiusServer type webvpn
ciscoasa(config)# tunnel-group RadiusServer general-attributes
ciscoasa(config-tunnel-general)# authentication server-group RADIUS
ciscoasa(config-tunnel-general)# accounting-server-group RADIUS
ciscoasa(config-tunnel-general)# tunnel-group RadiusServer webvpn-attributes
ciscoasa(config-tunnel-webvpn)# group-alias "Cisco Remote Access" enable
ciscoasa(config-tunnel-webvpn)# group-url http://www.example.com enable
ciscoasa(config-tunnel-webvpn)# group-url http://192.168.10.10 enable
ciscoasa(config-tunnel-webvpn)#
```

For a more extensive example, see Customizing Login Windows for Users of Clientless SSL VPN Sessions, page 4-27.

**Step 6** To exempt certain users from running Cisco Secure Desktop on a per connection profile basis if they enter one of the group-urls, enter the following command:

```
ciscoasa(config-tunnel-webvpn)# without-csd
ciscoasa(config-tunnel-webvpn)#
```

**Note** Entering this command prevents the detection of endpoint conditions for these sessions, so you may need to adjust the dynamic access policy (DAP) configuration.

**Step 7** To specify the DNS server group to use for a connection profile for clientless SSL VPN sessions, use the **dns-group** command. The group you specify must be one you already configured in global configuration mode (using the **dns server-group** and **name-server** commands).

By default, the connection profile uses the DNS server group *DefaultDNS*. However, this group must be configured before the security appliance can resolve DNS requests.

The following example configures a new DNS server group named *corp_dns* and specifies that server group for the connection profile *telecommuters*:

```
hostname(config)# dns server-group corp_dns
hostname(config-dns-server-group)# domain-name cisco.com
hostname(config-dns-server-group)# name-server 209.165.200.224

hostname(config)# tunnel-group telecommuters webvpn-attributes
hostname(config-tunnel-webvpn)# dns-group corp_dns
hostname(config-tunnel-webvpn)#
```

**Step 8** (Optional) To enable extracting a username from a client certificate for use in authentication and authorization, use the **pre-fill-username** command in tunnel-group webvpn-attributes mode. There is no default value.

```
ciscoasa(config)# pre-fill-username {ssl-client | clientless}
```

The **pre-fill-username** command enables the use of a username extracted from the certificate field specified in the **username-from-certificate** command (in tunnel-group general-attributes mode) as the username for username/password authentication and authorization. To use this pre-fill username from certificate feature, you must configure both commands.

**Note** In Version 8.0.4, the username is not pre-filled; instead, any data sent in the username field is ignored.

The following example, entered in global configuration mode, creates an IPsec remote access tunnel group named remotegrp, enables getting the username from a certificate, and specifies that the name for an authentication or authorization query for an SSL VPN client must be derived from a digital certificate:

```
ciscoasa(config)# tunnel-group remotegrp type ipsec_ra
ciscoasa(config)# tunnel-group remotegrp general-attributes
ciscoasa(config-tunnel-general)# username-from-certificate CN OU
ciscoasa(config)# tunnel-group remotegrp webvpn-attributes
ciscoasa(config-tunnel-webvpn)# pre-fill-username ssl-client
ciscoasa(config-tunnel-webvpn)#
```

**Step 9** (Optional) To specify whether to override the group policy or username attributes configuration for downloading an AnyConnect or SSL VPN client, use the **override-svc-download** command. This feature is disabled by default.

The security appliance allows clientless or AnyConnect client connections for remote users based on whether clientless and/or SSL VPN is enabled in the group policy or username attributes with the **vpn-tunnel-protocol** command. The **anyconnect ask** command further modifies the client user experience by prompting the user to download the client or return to the WebVPN home page.

However, you might want clientless users logging in under specific tunnel groups to not experience delays waiting for the download prompt to expire before being presented with the clientless SSL VPN home page. You can prevent delays for these users at the connection profile level with the **override-svc-download** command. This command causes users logging through a connection profile to be immediately presented with the clientless SSL VPN home page regardless of the **vpn-tunnel-protocol** or **anyconnect ask** command settings.

In the following example, the you enter tunnel-group webvpn attributes configuration mode for the connection profile *engineering* and enable the connection profile to override the group policy and username attribute settings for client download prompts:

```
hostname(config)# tunnel-group engineering webvpn-attributes
hostname(config-tunnel-webvpn)# override-svc-download
```

**Step 10** (Optional) To enable the display of a RADIUS reject message on the login screen when authentication is rejected, use the **radius-eject-message** command.

The following example enables the display of a RADIUS rejection message for the connection profile named engineering:

```
hostname(config)# tunnel-group engineering webvpn-attributes
hostname(config-tunnel-webvpn)# radius-reject-message
```

# Customizing Login Windows for Users of Clientless SSL VPN Sessions

You can set up different login windows for different groups by using a combination of customization profiles and connection profiles. For example, assuming that you had created a customization profile called salesgui, you can create a connection profile for clientless SSL VPN sessions called sales that uses that customization profile, as the following example shows:

**Step 1** In webvpn mode, define a customization for clientless SSL VPN access, in this case named salesgui and change the default logo to mycompanylogo.gif. You must have previously loaded mycompanylogo.gif onto the flash memory of the ASA and saved the configuration. See Chapter 14, “Introduction to Clientless SSL VPN” for details.

```
ciscoasa# webvpn
```

```
hostname (config-webvpn)# customization value salesgui
ciscoasa(config-webvpn-custom)# logo file disk0:\mycompanylogo.gif
ciscoasa(config-webvpn-custom)#
```

**Step 2** In global configuration mode, set up a username and associate with it the customization for clientless SSL VPN that you have just defined:

```
ciscoasa# username seller attributes
ciscoasa(config-username)# webvpn
ciscoasa(config-username-webvpn)# customization value salesgui
ciscoasa(config-username-webvpn)# exit
ciscoasa(config-username)# exit
ciscoasa#
```

**Step 3** In global configuration mode, create a tunnel-group for clientless SSL VPN sessions named sales:

```
ciscoasa# tunnel-group sales type webvpn
ciscoasa(config-tunnel-webvpn)#
```

**Step 4** Specify that you want to use the salesgui customization for this connection profile:

```
ciscoasa# tunnel-group sales webvpn-attributes
ciscoasa(config-tunnel-webvpn)# customization salesgui
```

**Step 5** Set the group URL to the address that the user enters into the browser to log in to the ASA; for example, if the ASA has the IP address 192.168.3.3, set the group URL to https://192.168.3.3:

```
ciscoasa(config-tunnel-webvpn)# group-url https://192.168.3.3.
ciscoasa(config-tunnel-webvpn)#
```

If a port number is required for a successful login, include the port number, preceded by a colon. The ASA maps this URL to the sales connection profile and applies the salesgui customization profile to the login screen that the user sees upon logging in to https://192.168.3.3.

# Configuring Microsoft Active Directory Settings for Password Management

**Note** If you are using an LDAP directory server for authentication, password management is supported with the Sun Microsystems JAVA System Directory Server (formerly named the Sun ONE Directory Server) and the Microsoft Active Directory.

- Sun—The DN configured on the ASA 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 ACI on the default password policy.
- Microsoft—You must configure LDAP over SSL to enable password management with Microsoft Active Directory.

See the general operations configuration guide for more information.

To use password management with Microsoft Active Directory, you must set certain Active Directory parameters as well as configuring password management on the ASA. This section describes the Active Directory settings associated with various password management actions. These descriptions assume that you have also enabled password management on the ASA and configured the corresponding password management attributes. The specific steps in this section refer to Active Directory terminology under Windows 2000 and include the following topics:

- Using Active Directory to Force the User to Change Password at Next Logon, page 4-29.
- Using Active Directory to Specify Maximum Password Age, page 4-30.
- Using Active Directory to Override an Account Disabled AAA Indicator, page 4-31
- Using Active Directory to Enforce Password Complexity, page 4-33.

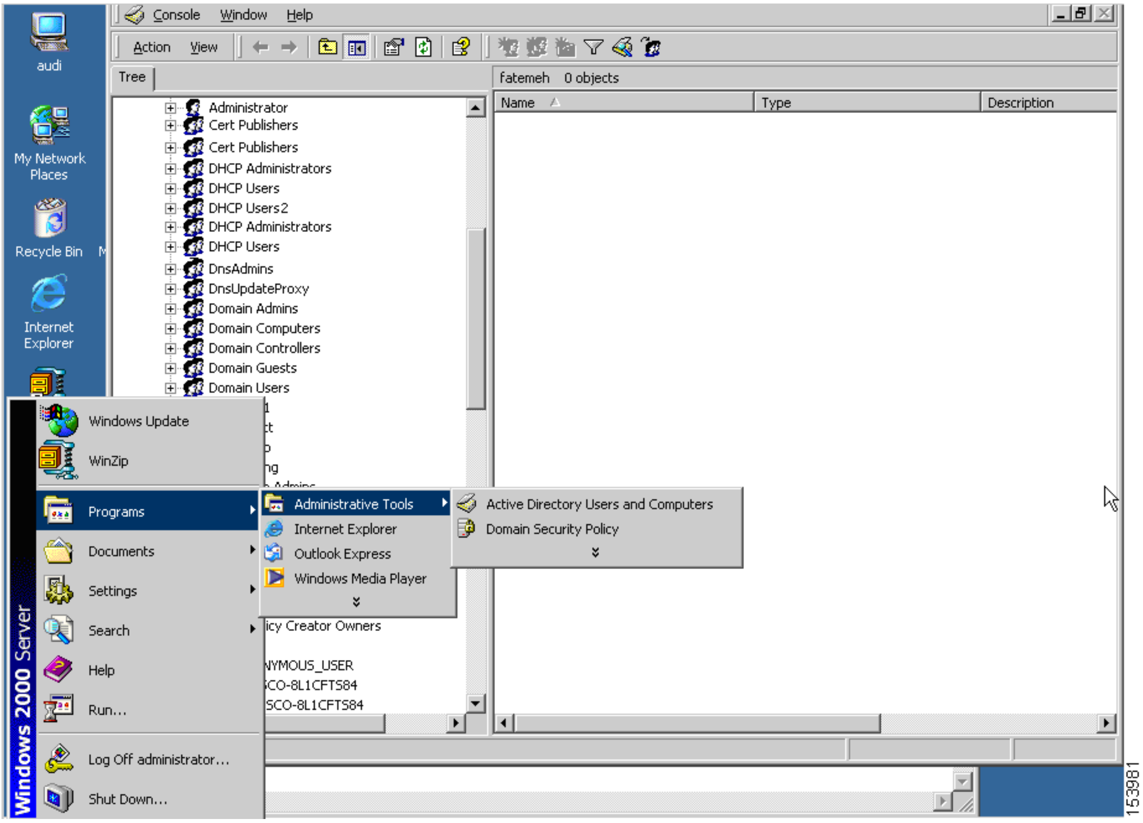
This section assumes that you are using an LDAP directory server for authentication.

## Using Active Directory to Force the User to Change Password at Next Logon

To force a user to change the user password at the next logon, specify the **password-management** command in tunnel-group general-attributes configuration mode on the ASA and perform the following steps under Active Directory:

**Step 1** Choose **Start > Programs > Administrative Tools > Active Directory Users and Computers** (Figure 4-1).

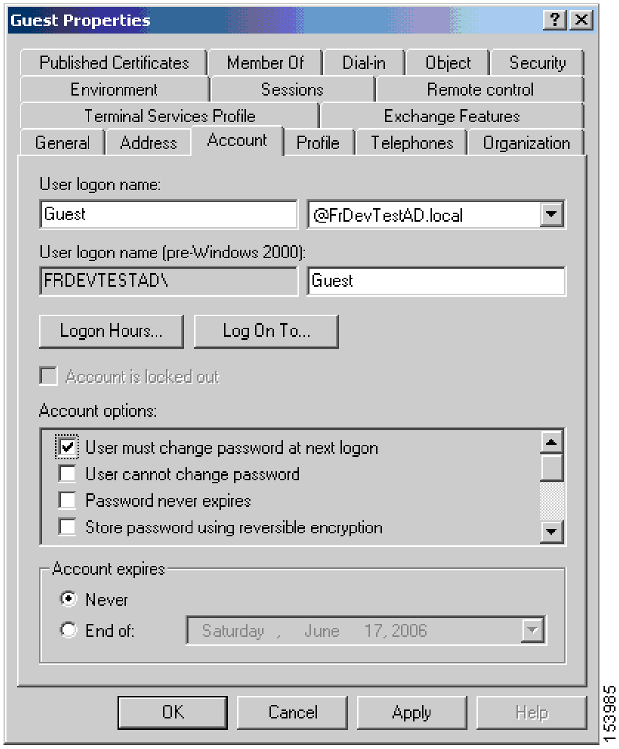
*Figure 4-1 Active Directory—Administrative Tools Menu*



**Step 2** Right-click to choose **Username > Properties > Account**.

**Step 3** Check the **User must change password at next logon** (Figure 4-2) check box.

*Figure 4-2 Active Directory—User Must Change Password at Next Logon*



User logon name: Guest @FrDevTestAD.local

User logon name (pre-Windows 2000): FRDEVTESTAD\ Guest

□ Account is locked out

Account options:

☑ User must change password at next logon
□ User cannot change password
□ Password never expires
□ Store password using reversible encryption

Account expires
● Never
○ End of: Saturday , June 17, 2006

The next time this user logs on, the ASA displays the following prompt: “New password required. Password change required. You must enter a new password with a minimum length *n* to continue.” You can set the minimum required password length, *n*, as part of the Active Directory configuration at Start > Programs > Administrative Tools > Domain Security Policy > Windows Settings > Security Settings > Account Policies > Password Policy. Select **Minimum password length**.

## Using Active Directory to Specify Maximum Password Age

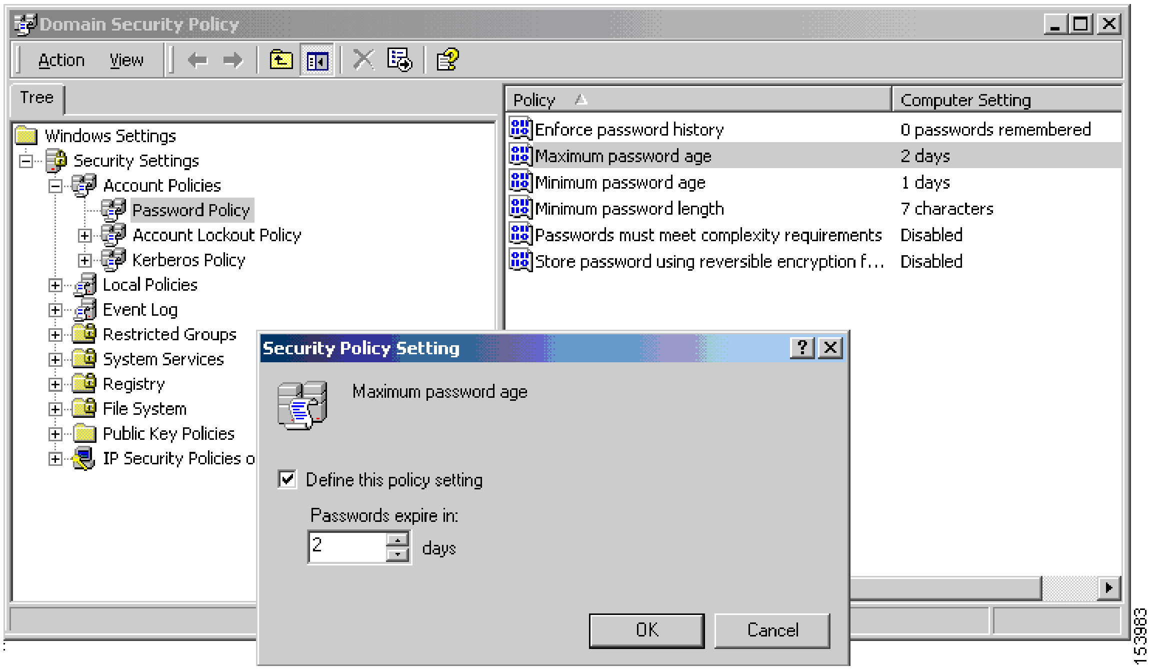
To enhance security, you can specify that passwords expire after a certain number of days. To specify a maximum password age for a user password, specify the **password-management** command in tunnel-group general-attributes configuration mode on the ASA and perform the following steps under Active Directory:

**Step 1** Choose **Start > Programs > Administrative Tools > Domain Security Policy > Windows Settings > Security Settings > Account Policies > Password Policy**.

**Step 2** Double-click Maximum password age. The Security Policy Setting dialog box appears.

**Step 3** Check the **Define this policy setting** check box and specify the maximum password age, in days, that you want to allow.

*Figure 4-3 Active Directory—Maximum Password Age*



| Policy | Computer Setting |
|---|---|
| Enforce password history | 0 passwords remembered |
| Maximum password age | 2 days |
| Minimum password age | 1 days |
| Minimum password length | 7 characters |
| Passwords must meet complexity requirements | Disabled |
| Store password using reversible encryption f... | Disabled |

Maximum password age

☑ Define this policy setting

Passwords expire in:

2 days

**Note** The **radius-with-expiry** command, formerly configured as part of tunnel-group remote-access configuration to perform the password age function, is deprecated. The **password-management** command, entered in tunnel-group general-attributes mode, replaces it.

## Using Active Directory to Override an Account Disabled AAA Indicator

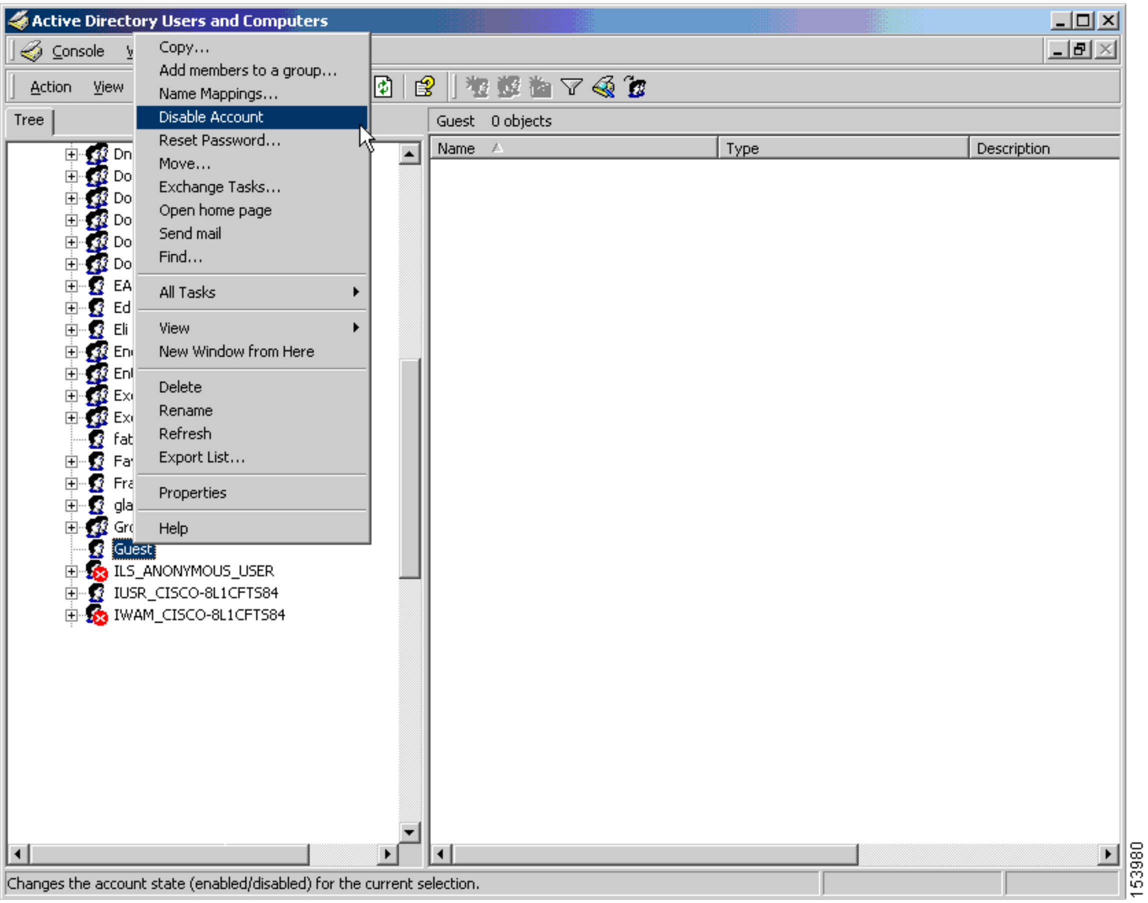
To override an account-disabled indication from a AAA server, use the **override-account-disable** command in tunnel-group general-attributes configuration mode on the ASA and perform the following steps under Active Directory.

**Note** Allowing override account-disabled is a potential security risk.

**Step 1** Select Start > Programs > Administrative Tools > Active Directory Users and Computers.

**Step 2** Right-click Username > Properties > Account and select Disable Account from the menu.

*Figure 4-4 Active Directory—Override Account Disabled*



Changes the account state (enabled/disabled) for the current selection.

The user should be able to log on successfully, even though a AAA server provides an account-disabled indicator.

## Using Active Directory to Enforce Minimum Password Length

To enforce a minimum length for passwords, specify the **password-management** command in tunnel-group general-attributes configuration mode on the ASA and perform the following steps under Active Directory:

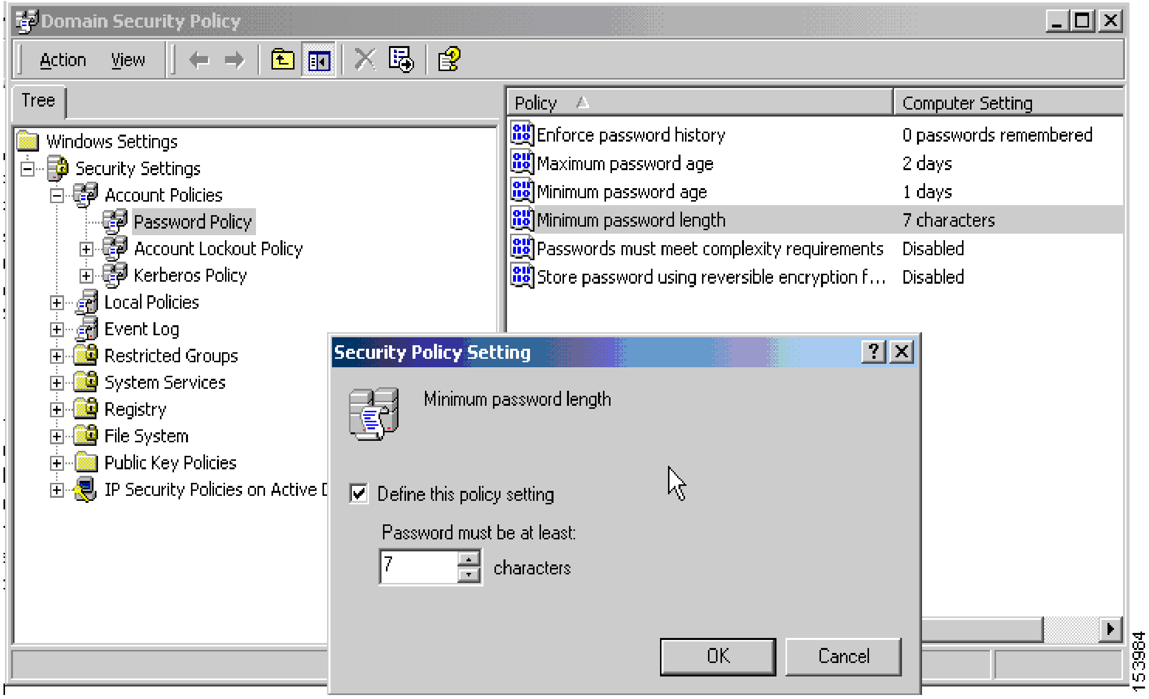
**Step 1** Select Start > Programs > Administrative Tools > Domain Security Policy.

**Step 2** Select Windows Settings > Security Settings > Account Policies > Password Policy.

**Step 3** Double-click Minimum Password Length. The Security Policy Setting dialog box appears.

**Step 4** Check the Define this policy setting check box and specify the minimum number of characters that the password must contain.

*Figure 4-5 Active Directory—Minimum Password Length*



- Windows Settings
  - Security Settings
    - Account Policies
      - Password Policy
      - Account Lockout Policy
      - Kerberos Policy
    - Local Policies
    - Event Log
    - Restricted Groups
    - System Services
    - Registry
    - File System
    - Public Key Policies
    - IP Security Policies on Active D

| Policy | Computer Setting |
|---|---|
| Enforce password history | 0 passwords remembered |
| Maximum password age | 2 days |
| Minimum password age | 1 days |
| Minimum password length | 7 characters |
| Passwords must meet complexity requirements | Disabled |
| Store password using reversible encryption f... | Disabled |

Minimum password length

☑ Define this policy setting

Password must be at least:

7 characters

## Using Active Directory to Enforce Password Complexity

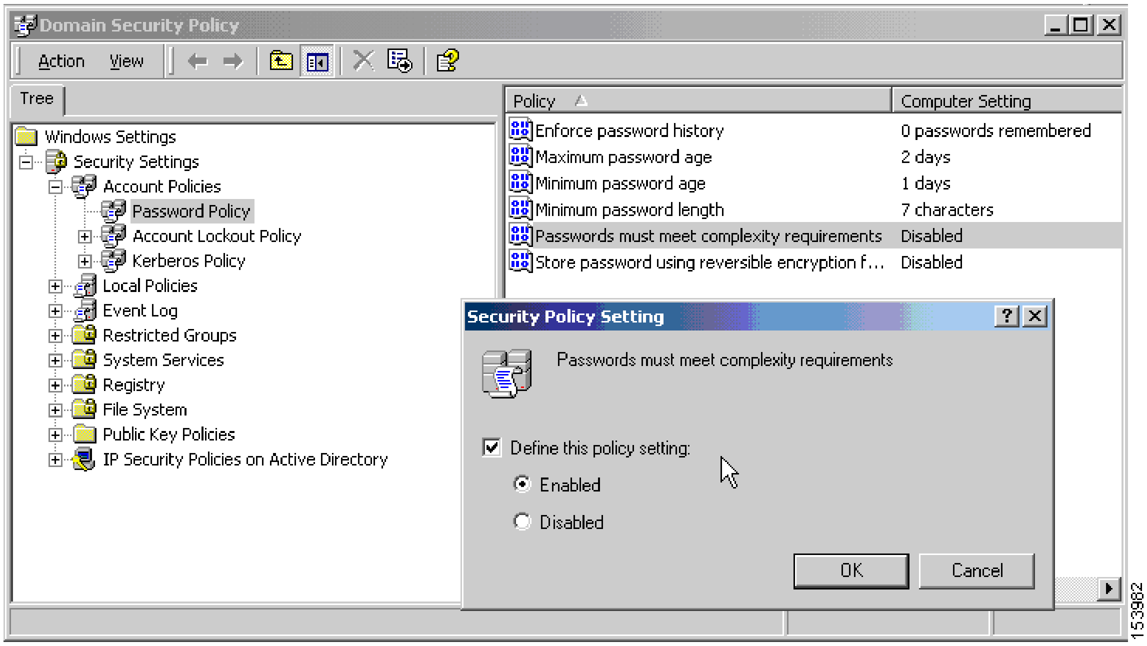
To enforce complex passwords—for example, to require that a password contain upper- and lowercase letters, numbers, and special characters—enter the **password-management** command in tunnel-group general-attributes configuration mode on the ASA and perform the following steps under Active Directory:

**Step 1** Select Start > Programs > Administrative Tools > Domain Security Policy. Select Windows Settings > Security Settings > Account Policies > Password Policy.

**Step 2** Double-click Password must meet complexity requirements to open the Security Policy Setting dialog box.

**Step 3** Check the Define this policy setting check box and select **Enable**.

*Figure 4-6 Active Directory—Enforce Password Complexity*



| Policy | Computer Setting |
|---|---|
| Enforce password history | 0 passwords remembered |
| Maximum password age | 2 days |
| Minimum password age | 1 days |
| Minimum password length | 7 characters |
| Passwords must meet complexity requirements | Disabled |
| Store password using reversible encryption f... | Disabled |

Security Policy Setting

Passwords must meet complexity requirements

☑ Define this policy setting:

● Enabled

○ Disabled

Enforcing password complexity takes effect only when the user changes passwords; for example, when you have configured Enforce password change at next login or Password expires in *n* days. At login, the user receives a prompt to enter a new password, and the system will accept only a complex password.

# Configuring the Connection Profile for RADIUS/SDI Message Support for the AnyConnect Client

This section describes procedures to ensure that the AnyConnect VPN client using RSA SecureID Software tokens can properly respond to user prompts delivered to the client through a RADIUS server proxying to an SDI server(s). This section contains the following topics:

- AnyConnect Client and RADIUS/SDI Server Interaction
- Configuring the Security Appliance to Support RADIUS/SDI Messages

**Note** If you have configured the double-authentication feature, SDI authentication is supported only on the primary authentication server.

## AnyConnect Client and RADIUS/SDI Server Interaction

When a remote user connects to the ASA with the AnyConnect VPN client and attempts to authenticate using an RSA SecurID token, the ASA communicates with the RADIUS server, which in turn, communicates with the SDI server about the authentication.

During authentication, the RADIUS server presents access challenge messages to the ASA. Within these challenge messages are reply messages containing text from the SDI server. The message text is different when the ASA is communicating directly with an SDI server than when communicating through the RADIUS proxy. Therefore, in order to appear as a native SDI server to the AnyConnect client, the ASA must interpret the messages from the RADIUS server.

Also, because the SDI messages are configurable on the SDI server, the message text on the ASA must match (in whole or in part) the message text on the SDI server. Otherwise, the prompts displayed to the remote client user may not be appropriate for the action required during authentication. The AnyConnect client may fail to respond and authentication may fail.

"Configuring the Security Appliance to Support RADIUS/SDI Messages" section on page 4-35 describes how to configure the ASA to ensure successful authentication between the client and the SDI server.

## Configuring the Security Appliance to Support RADIUS/SDI Messages

To configure the ASA to interpret SDI-specific RADIUS reply messages and prompt the AnyConnect user for the appropriate action, perform the following steps:

**Step 1** Configure a connection profile (tunnel group) to forward RADIUS reply messages in a manner that simulates direct communication with an SDI server using the **proxy-auth sdi** command from tunnel-group webvpn configuration mode. Users authenticating to the SDI server must connect over this connection profile.

For example:

```
hostname(config)# tunnel-group sales webvpn attributes
hostname(tunnel-group-webvpn)# proxy-auth sdi
```

**Step 2** Configure the RADIUS reply message text on the ASA to match (in whole or in part) the message text sent by the RADIUS server with the **proxy-auth_map sdi** command from tunnel-group webvpn configuration mode.

The default message text used by the ASA is the default message text used by Cisco Secure Access Control Server (ACS). If you are using Cisco Secure ACS, and it is using the default message text, you do not need to configure the message text on the ASA. Otherwise, use the **proxy-auth_map sdi** command to ensure the message text matches.

Table 4-3 shows the message code, the default RADIUS reply message text, and the function of each message. Because the security appliance searches for strings in the order that they appear in the table, you must ensure that the string you use for the message text is not a subset of another string.

For example, "new PIN" is a subset of the default message text for both new-pin-sup and next-ccode-and-reauth. If you configure new-pin-sup as "new PIN", when the security appliance receives "new PIN with the next card code" from the RADIUS server, it will match the text to the new-pin-sup code instead of the next-ccode-and-reauth code.

*Table 4-3 SDI Op-codes, Default Message Text, and Message Function*

| Message Code | Default RADIUS Reply Message Text | Function |
|---|---|---|
| next-code | Enter Next PASSCODE | Indicates the user must enter the NEXT tokencode without the PIN. |
| new-pin-sup | Please remember your new PIN | Indicates the new system PIN has been supplied and displays that PIN for the user. |

| Message Code | Default RADIUS Reply Message Text | Function |
|---|---|---|
| new-pin-meth | Do you want to enter your own pin | Requests from the user which new PIN method to use to create a new PIN. |
| new-pin-req | Enter your new Alpha-Numerical PIN | Indicates a user-generated PIN and requests that the user enter the PIN. |
| new-pin-reenter | Reenter PIN: | Used internally by the ASA for user-supplied PIN confirmation. The client confirms the PIN without prompting the user. |
| new-pin-sys-ok | New PIN Accepted | Indicates the user-supplied PIN was accepted. |
| next-ccode-and-reauth | new PIN with the next card code | Follows a PIN operation and indicates the user must wait for the next tokencode and to enter both the new PIN and next tokencode to authenticate. |
| ready-for-sys-pin | ACCEPT A SYSTEM GENERATED PIN | Used internally by the ASA to indicate the user is ready for the system-generated PIN. |

The following example enters aaa-server-host mode and changes the text for the RADIUS reply message new-pin-sup*:*

```
hostname(config)# aaa-server radius_sales host 10.10.10.1
hostname(config-aaa-server-host)# proxy-auth_map sdi new-pin-sup “This is your new PIN”
```

# Group Policies

This section describes group policies and how to configure them. It includes the following topics:

- Default Group Policy, page 4-37
- Configuring Group Policies, page 4-39

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 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’s internal database. External groups are configured on an external authentication server, such as RADIUS. Group policies include the following attributes:

- Identity
- Server definitions
- Client firewall settings
- Tunneling protocols
- IPsec settings

- Hardware client settings
- Filters
- Client configuration settings
- Connection settings

## Default Group Policy

The ASA 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, but this default group policy does not take effect unless you configure the ASA 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:

```
ciscoasa(config)# show running-config all group-policy DfltGrpPolicy
hostname(config)#
```

To configure the default group policy, enter the following command:

```
ciscoasa(config)# group-policy DfltGrpPolicy internal
hostname(config)#
```

**Note** The default group policy is always internal. Despite the fact that the command syntax is ciscoasa(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:

```
ciscoasa(config)# group-policy DfltGrpPolicy attributes
```

**Note** The attributes mode applies only to internal group policies.

The default group policy, DfltGrpPolicy, that the ASA provides is as follows:

```
hostname# show run all group-policy DfltGrpPolicy
group-policy DfltGrpPolicy internal
group-policy DfltGrpPolicy attributes
 banner none
 wins-server none
 dns-server value 10.10.10.1.1
 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
 vpn-tunnel-protocol ikev1 ikev2 l2tp-ipsec ssl-client ssl-clientless
 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
ipv6-split-tunnel-policy tunnelall
split-tunnel-network-list none
default-domain value cisco.com
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
client-bypass-protocol disable
gateway-fqdn none
leap-bypass disable
nem disable
backup-servers keep-client-config
msie-proxy server none
msie-proxy method no-modify
msie-proxy except-list none
msie-proxy local-bypass disable
msie-proxy pac-url none
msie-proxy lockdown enable
vlan none
nac-settings none
address-pools none
ipv6-address-pools none
smartcard-removal-disconnect enable
scep-forwarding-url none
client-firewall none
client-access-rule none
webvpn
 url-list none
 filter none
 homepage none
 html-content-filter none
 port-forward name Application Access
 port-forward disable
 http-proxy disable
 sso-server none
 anyconnect ssl dtls enable
 anyconnect mtu 1406
 anyconnect firewall-rule client-interface private none
 anyconnect firewall-rule client-interface public none
 anyconnect keep-installer installed
 anyconnect ssl keepalive 20
 anyconnect ssl rekey time none
 anyconnect ssl rekey method none
 anyconnect dpd-interval client 30
 anyconnect dpd-interval gateway 30
 anyconnect ssl compression none
 anyconnect dtls compression lzs
 anyconnect modules none
 anyconnect profiles none
 anyconnect ask none
 customization none
 keep-alive-ignore 4
 http-comp gzip
 download-max-size 2147483647
 upload-max-size 2147483647
 post-max-size 2147483647
```

```
  user-storage none
  storage-objects value cookies,credentials
  storage-key none
  hidden-shares none
  smart-tunnel disable
  activex-relay enable
  unix-auth-uid 65534
  unix-auth-gid 65534
  file-entry enable
  file-browsing enable
  url-entry enable
  deny-message value Login was successful, but because certain criteria have not been met
or due to some specific group policy, you do not have permission to use any of the VPN
features. Contact your IT administrator for more information
  smart-tunnel auto-signon disable
  anyconnect ssl df-bit-ignore disable
  anyconnect routing-filtering-ignore disable
  smart-tunnel tunnel-policy tunnelall
  always-on-vpn profile-setting
```

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.

You can perform these configuration tasks in both single context mode or multiple-context mode:

**Note** Multiple-context mode applies only to IKEv2 and IKEv1 site to site and does not apply to AnyConnect, Clientless SSL VPN, legacy Cisco VPN client, the Apple native VPN client, the Microsoft native VPN client, or cTCP for IKEv1 IPsec.

## 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 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 refer to user names on the RADIUS server. In other words, if you configure external group X on the ASA, 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. 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 supports user authorization on an external LDAP or RADIUS server. Before you configure the ASA to use an external server, you must configure the server with the correct ASA authorization attributes and, from a subset of these attributes, assign specific permissions to individual users. Follow the instructions in Appendix 13, "Configuring an External Server for Authorization and Authentication" to configure your external server.

To configure an external group policy, perform 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)#
```

**Note** You can configure several vendor-specific attributes (VSAs), as described in Appendix 13, "Configuring an External Server for Authorization and Authentication". If a RADIUS server is configured to return the Class attribute (#25), the ASA 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—for example, OU=Finance.

## Creating an Internal Group Policy

To configure an internal group policy, enter configuration mode, use the group-policy command, specify a name, and the **internal** type for the group policy:

```
hostname(config)# group-policy group_policy_name internal
hostname(config)#
```

For example, the following command creates the internal group policy named GroupPolicy1:

```
hostname(config)# group-policy GroupPolicy1 internal
hostname(config)#
```

**Note** You cannot change the name of a group policy after you create it.

You can conifgure the attributes of an internal group policy by copying 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
ciscoasa(config-group-policy)#
```

For example, the following command creates the internal group policy named GroupPolicy2 by copying the attributes of GroupPolicy1:

```
hostname(config)# group-policy GroupPolicy2 internal from GroupPolicy1
ciscoasa(config-group-policy)#
```

# Configuring General Internal Group Policy Attributes

## Group Policy Name

The group policy name was chosen when the internal group policy was created. You cannot change the name of a group policy once it has been created. See Creating an Internal Group Policy, page 4-40 for more information.

## Configuring the Group Policy Banner Message

Specify the banner, or welcome message, if any, that you want to display. The default is no banner. The message that you specify is displayed on remote clients when they connect. To specify a banner, enter the **banner** command in group-policy configuration mode. The banner text can be up to 510 characters long. Enter the “\n” sequence to insert a carriage return.

**Note** A carriage-return and line-feed included in the banner counts as two characters.

To delete a banner, enter the **no** form of this command. Be aware that using the **no** version of the command deletes all banners for the 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 a value for the banner string, as follows:

```
ciscoasa(config-group-policy)# banner {value banner_string | none}
```

The following example shows how to create a banner for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# banner value Welcome to Cisco Systems ASA 9.0.
```

## Specifying Address Pools for Remote Access Connections

When remote access clients connect to the ASA, the ASA can assign the client an IPv4 or IPv6 address based on the group-policy specified for the connection.

You can specify a list of up to six local address pools to use for local address allocation. The order in which you specify the pools is significant. The ASA allocates addresses from these pools in the order in which the pools appear in this command.

## Assigning an IPv4 Address Pool to an Internal Group Policy

### Prerquisite

Create the IPv4 address pool. See Chapter 5, “Configuring IP Addresses for VPNs.”

### Detailed Steps

| | Command | Purpose |
|---|---|---|
| **Step 1** | `group-policy` *`value`* `attributes`<br>**Example:**<br>`ciscoasa>` **`en`**<br>`ciscoasa#` **`config t`**<br>`ciscoasa(config)#` **`group-policy FirstGroup attributes`**<br>`ciscoasa(config-group-policy)#` | Enter group policy configuration mode. |
| **Step 2** | **`address-pools value`** *`pool-name1 pool-name2 pool-name6`*<br>**Example:**<br>`asa4(config-group-policy)# address-pools value ipv4-pool1 ipv4-pool2 ipv4-pool3`<br>`asa4(config-group-policy)#` | Assigns the address pool named ipv4-pool1, ipv4-pool2, and ipv4pool3 to the FirstGroup group policy.<br>You are allowed to specify up to 6 address pools for group-policy. |
| **Step 3** | (Optional)<br>**no address-pools value pool-name1 pool-name2 pool-name6**<br>**Example:**<br>`ciscoasa(config-group-policy)#` **`no address-pools value ipv4-pool1 ipv4-pool2 ipv4-pool3`**<br>`ciscoasa(config-group-policy)#` | Use the **`no address-pools value pool-name`** command to remove the address-pools from the goup policy configuration and returns the address pool setting to inherit the address pool information from other sources such as the DefltGroupPolicy. |
| **Step 4** | (Optional)<br>**address-pools none**<br>**Example:**<br>`ciscoasa(config-group-policy)#` **`address-pools none`**<br>`ciscoasa(config-group-policy)#` | The **address-pools none** command disables this attribute from being inherited from other sources of policy, such as the DefltGrpPolicy: |
| **Step 5** | (Optional)<br>**no address-pools none**<br>**Example:**<br>`ciscoasa(config-group-policy)#` **`no address-pools none`**<br>`ciscoasa(config-group-policy)#` | The **no address pools none** command removes the **address-pools none** command from the group policy, restoring the default value, which is to allow inheritance. |

## Assigning an IPv6 Address Pool to an Internal Group Policy

### Prerquisite

Create the IPv6 address pool. See Chapter 5, “Configuring IP Addresses for VPNs.”

### Detailed Steps

| | Command | Purpose |
|---|---|---|
| **Step 1** | group-policy *value* attributes<br>**Example:**<br>ciscoasa> **en**<br>ciscoasa# **config t**<br>ciscoasa(config)# **group-policy FirstGroup attributes**<br>ciscoasa(config-group-policy)# | Enter group policy configuration mode. |
| **Step 2** | **ipv6-address-pools value** *pool-name1 pool-name2 pool-name6*<br>**Example:**<br>ciscoasa(config-group-policy)# **ipv6-address-pools value ipv6-pool1 ipv6-pool2 ipv6-pool3**<br>ciscoasa(config-group-policy)# | Assigns the address pool named ipv6-pool to the FirstGroup group policy.<br>You can assign up to six ipv6 address pools to a group policy.<br>This example shows ipv6-pool1, ipv6-pool2, and ipv6-pool3 being assigned to the FirstGroup group policy. |
| **Step 3** | (Optional)<br>**no ipv6-address-pools value pool-name1 pool-name2 pool-name6**<br>**Example:**<br>ciscoasa(config-group-policy)# **no ipv6-address-pools value ipv6-pool1 ipv6-pool2 ipv6-pool3**<br>ciscoasa(config-group-policy)# | Use the **no ipv6-address-pools value pool-name** command to remove the address-pools from the goup policy configuration and returns the address pool setting to inherit the address pool information from other sources such as the DfltGroupPolicy. |
| **Step 4** | (Optional)<br>**ipv6-address-pools none**<br>**Example:**<br>ciscoasa(config-group-policy)# **ipv6-address-pools none**<br>ciscoasa(config-group-policy)# | The **ipv6-address-pools none** command disables this attribute from being inherited from other sources of policy, such as the DfltGrpPolicy: |
| **Step 5** | (Optional)<br>**no ipv6-address-pools none**<br>**Example:**<br>ciscoasa(config-group-policy)# **no ipv6-address-pools none**<br>ciscoasa(config-group-policy)# | The **no ipv6-address pools none** command removes the **ipv6-address-pools none** command from the group policy, restoring the default value, which is to allow inheritance. |

## Specifying the Tunneling Protocol for the Group Policy

Specify the VPN tunnel type for this group policy by entering the **vpn-tunnel-protocol {ikev1 | ikev2 | l2tp-ipsec | ssl-client | ssl-clientless**} command from group-policy configuration mode.

The default value is to inherit the attributes of the Default Group Policy. 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.
- **l2tp-ipsec**—Negotiates an IPsec tunnel for an L2TP connection.
- **ssl-client**—Negotiates an SSL tunnel using TLS or DTLS with the AnyConnect Secure Mobility Client.
- **ssl-clientless**—Provides VPN services to remote users via an HTTPS-enabled web browser, and does not require a client.

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:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# vpn-tunnel-protocol ikev1
ciscoasa(config-group-policy)#
```

## Specifying a VLAN for Remote Access or Applying a Unified Access Control Rule to the Group Policy

Filters consist of rules that determine whether to allow or reject tunneled data packets coming through the ASA, based on criteria such as source address, destination address, and protocol. You can specify an IPv4 or IPv6 unified access control list for your group policy or allow it to inherit the ACLs specified in the Default Group Policy.

Choose one of the following options to specify an egress VLAN (also called “VLAN mapping”) for remote access or specify an ACL to filter the traffic:

- Enter the following command in group-policy configuration mode to specify the egress VLAN for remote access VPN sessions assigned to this group policy or to a group policy that inherits this group policy:

  ```
  ciscoasa(config-group-policy)# [no] vlan {vlan_id |none}
  ```

  *no vlan* removes the *vlan_id* from the group policy. The group policy inherits the vlan value from the default group policy.

  *none* removes the *vlan_id* from the group policy and disables VLAN mapping for this group policy. The group policy does not inherit the vlan value from the default group policy.

  *vlan_id* is the number of the VLAN, in decimal format, to assign to remote access VPN sessions that use this group policy. The VLAN must be configured on this ASA per the instructions in the “Configuring VLAN Subinterfaces and 802.1Q Trunking” section on page 11-36 in the general operations configuration guide.

  **Note** The egress VLAN feature works for HTTP connections, but not for FTP and CIFS.

- Specify the name of the access control rule (ACL) to apply to VPN session, using the **vpn-filter** command in group policy mode. You can specify an IPv4 or IPv6 ACL using the vpn-filter command.

**Note** In previous releases, the deprecated ipv6-vpn-filter command could be used to specify an IPv6 ACL if there were no IPv6 entries specified by vpn-filter. As of ASA 9.1(4), ipv6-vpn-filter has been disabled and IPv6 ACL entries must be specified using the vpn-filter command. If ipv6-vpn-filter is set, the VPN connection will be terminated.

**Note** You can also configure this attribute in username mode, in which case the value configured under username supersedes the group-policy value.

```
ciscoasa(config-group-policy)# vpn-filter {value ACL name | none}
ciscoasa(config-group-policy)#
```

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 ACL and sets a null value, thereby disallowing an ACL.

The following example shows how to set a filter that invokes an ACL named acl_vpn for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# vpn-filter acl_vpn
ciscoasa(config-group-policy)#
```

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:

```
ciscoasa(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:

```
ciscoasa(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** allows 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 10.10.10.1 255.255.255.255 eq 23 192.168.1.0 255.255.255.0** allows 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 remote network to telnet to the local network:

```
ciscoasa(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:

```
ciscoasa(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** allows 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** allows the remote network to initiate a connection to the local network on any TCP port if it uses a source port of 23.

## Specifying a NAC Policy for a Group Policy

This command selects the name of a Network Admission Control policy to apply to this group policy. You can assign an optional NAC policy to each group policy. The default value is --None--.

### Prerquisite

Create a NAC policy. See Configuring Network Admission Control, page 7-1.

### Detailed Steps

| | Command | Purpose |
|---|---|---|
| **Step 1** | `group-policy` *`value`* `attributes`<br>**Example:**<br>`ciscoasa>` **`en`**<br>`ciscoasa#` **`config t`**<br>`ciscoasa(config)#` **`group-policy FirstGroup attributes`**<br>`ciscoasa(config-group-policy)#` | Enter group policy configuration mode. |
| **Step 2** | **`nac-settings value`** *`nac-policy-name`*<br>**Example:**<br>ciscoasa(config-group-policy)# **`nac-settings value nac-policy-1`**<br>ciscoasa(config-group-policy)# | Assigns the NAC policy named nac-policy-1 to the FirstGroup group policy. |

## Specifying VPN Access Hours for a Group Policy

### Prerequisites

Create a time range. See the "Configuring Time Ranges" section on page 20-15 in the general operations configuration guide.

### Detailed Steps

| | Command | Purpose |
|---|---|---|
| **Step 1** | `group-policy` *`value`* `attributes`<br>**Example:**<br>`ciscoasa>` **`en`**<br>`ciscoasa#` **`config t`**<br>`ciscoasa(config)#` **`group-policy FirstGroup attributes`**<br>`ciscoasa(config-group-policy)#` | Enter group policy configuration mode. |
| **Step 2** | `ciscoasa(config-group-policy)#` **`vpn-access-hours value`** `{`*`time-range-name`* **`\|`** **`none`**`}`<br>**Example:**<br>`ciscoasa(config-group-policy)#` **`vpn-access-hours value business-hours`**<br>ciscoasa(config-group-policy)# | You can set the VPN access hours by associating a configured time-range policy with a group policy using the **vpn-access-hours** command in group-policy configuration mode.<br>This command assigns a VPN access time range named business-hours to the group policy named FirstGroup.<br>A group policy can inherit a time-range value from a default or specified group policy. To prevent this inheritance, enter the **none** keyword instead of the name of a time-range in this command. This keyword sets VPN access hours to a null value, which allows no time-range policy. |

## Specifying Simultaneous VPN Logins for a Group Policy

Specify the number of simultaneous logins allowed for any user, using the **vpn-simultaneous-logins** command in group-policy configuration mode.

```
ciscoasa(config-group-policy)# vpn-simultaneous-logins integer
```

The default value is 3. The range is an integer in the range 0 through 2147483647. A group policy can inherit this value from another group policy. Enter 0 to disable login and prevent user access. The following example shows how to allow a maximum of 4 simultaneous logins for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# vpn-simultaneous-logins 4
ciscoasa(config-group-policy)#
```

**Note** While the maximum limit for the number of simultaneous logins is very large, allowing several simultaneous logins could compromise security and affect performance.

Stale AnyConnect, IPsec Client, or Clientless sessions (sessions that are terminated abnormally) might remain in the session database, even though a "new" session has been established with the same username.

If the value of vpn-simultaneous-logins is 1, and the same user logs in again after an abnormal termination, then the stale session is removed from the database and the new session is established. If, however, the existing session is still an active connection and the same user logs in again, perhaps from another PC, the first session is logged off and removed from the database, and the new session is established.

If the number of simultaneous logins is a value greater than 1, then, when you have reached that maximum number and try to log in again, the session with the longest idle time is logged off. If all current sessions have been idle an equally long time, then the oldest session is logged off. This action frees up a session and allows the new login.

## Restricting Access to a Specific Connection Profile

Specify whether to restrict remote users to access only through the connection profile, using the **group-lock** command in group-policy configuration mode.

```
ciscoasa(config-group-policy)# group-lock {value tunnel-grp-name | none}
ciscoasa(config-group-policy)# no group-lock
ciscoasa(config-group-policy)#
```

The *tunnel-grp-name* variable specifies the name of an existing connection profile that the ASA requires for the user to connect. Group-lock restricts users by checking if the group configured in the VPN client is the same as the connection profile to which the user is assigned. If it is not, the ASA prevents the user from connecting. If you do not configure group-lock, the ASA authenticates users without regard to the assigned group. Group locking is disabled by default.

To remove the **group-lock** attribute from the running configuration, enter the **no** form of this command. This option allows inheritance of a value from another group policy.

To disable group-lock, enter the **group-lock** command with the **none** keyword. The none keyword sets group-lock to a null value, thereby allowing no group-lock restriction. It also prevents inheriting a group-lock value from a default or specified group policy

## Specifying the Maximum VPN Connection Time in a Group Policy

**Step 1** Configure a maximum amount of time for VPN connections, using the **vpn-session-timeout** command in group-policy configuration mode or in username configuration mode.

```
ciscoasa(config-group-policy)# vpn-session-timeout {minutes | none}
ciscoasa(config-group-policy)#
```

The minimum time is 1 minute, and the maximum time is 35791394 minutes. There is no default value. At the end of this period of time, the ASA 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. Specifying the **none** keyword permits an unlimited session timeout period and sets session timeout with a null value, which disallows a session timeout.

The following example shows how to set a VPN session timeout of 180 minutes for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# vpn-session-timeout 180
ciscoasa(config-group-policy)#
```

**Step 2** Configure the the time at which a session-timeout alert message is displayed to the user using the **vpn-session-timeout alert-interval** {minutes | none} command. This alert message tells users how many minutes left they have until their VPN session is automatically disconnected.

The following example shows how to set the **vpn-session-timeout alert-interval** so that users will be notified 20 minutes before their VPN session is disconnected. You can specify a range of 1-30 minutes.

```
hostname(config-webvpn)# vpn-session-timeout alert-interval 20
```

The *none* parameter indicates that users will not receive an alert.

Use the no form of the command to indicate that the VPN session timeout alert-interval attribute will be inherited from the Default Group Policy:

```
no vpn-session-timeout alert-interval
```

## Specifying a VPN Session Idle Timeout for a Group Policy

**Step 1** Configure the user timeout period by entering the **vpn-idle-timeout** command in group-policy configuration mode or in username configuration mode:

```
ciscoasa(config-group-policy)# vpn-idle-timeout {minutes | none}
ciscoasa(config-group-policy)#
```

AnyConnect (SSL IPsec/IKEv2): Use the global WebVPN default-idle-timeout value (seconds) from the command: **ciscoasa(config-webvpn)# default-idle-timeout**

The range for this value in the WebVPN **default-idle-timeout** command is 60-86400 seconds; the default Global WebVPN Idle timeout in seconds -- default is 1800 seconds (30 min).

**Note** A non-zero idle timeout value is required by ASA for all AnyConnect connections.

For a WebVPN user, the **default-idle-timeout** value is enforced only if vpn-idle-timeout none is set in the group policy/username attribute.

Site-to-Site (IKEv1, IKEv2) and IKEv1 remote-access: Disable timeout and allow for an unlimited idle period.

The following example shows how to set a VPN idle timeout of 15 minutes for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# vpn-idle-timeout 15
ciscoasa(config-group-policy)#
```

**Step 2** Configure the 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 indicates that users will not receive an alert.

Use the no form of the command to indicate that the VPN idle timeout alert-interval attribute will be inherited from the Default Group Policy:

```
no vpn-idle-timeout alert-interval
```

## Configuring WINS and DNS Servers for a Group Policy

You can specify primary and secondary WINS servers and DNS servers. The default value in each case is none. To specify these servers, perform the following steps:

**Step 1** Specify the primary and secondary WINS servers:

```
hostname(config-group-policy)# wins-server value {ip_address [ip_address] | none}
ciscoasa(config-group-policy)#
```

The first IP address specified is that of the primary WINS server. The second (optional) IP address is that of the secondary WINS server. Specifying the **none** keyword instead of an IP address sets WINS servers to a null value, which allows no WINS servers and prevents inheriting a value from a default or specified group policy.

Every time that you enter the **wins-server** command, you overwrite the existing setting. For example, if you configure WINS server x.x.x.x and then configure WINS server y.y.y.y, the second command overwrites the first, and y.y.y.y becomes the sole WINS server. The same is true for multiple servers. To add a WINS server rather than overwrite previously configured servers, include the IP addresses of all WINS servers when you enter this command.

The following example shows how to configure WINS servers with the IP addresses 10.10.10.15 and 10.10.10.30 for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# wins-server value 10.10.10.15 10.10.10.30
ciscoasa(config-group-policy)#
```

**Step 2** Specify the primary and secondary DNS servers:

```
hostname(config-group-policy)# dns-server value {ip_address [ip_address] | none}
ciscoasa(config-group-policy)#
```

The first IP address specified is that of the primary DNS server. The second (optional) IP address is that of the secondary DNS server. Specifying the **none** keyword instead of an IP address sets DNS servers to a null value, which allows no DNS servers and prevents inheriting a value from a default or specified group policy. You can specify up to four DNS server addresses: Up to two IPv4 addresses and two IPv6 addreses.

Every time that you enter the **dns-server** command you overwrite the existing setting. For example, if you configure DNS server x.x.x.x and then configure DNS server y.y.y.y, the second command overwrites the first, and y.y.y.y becomes the sole DNS server. The same is true for multiple servers. To add a DNS server rather than overwrite previously configured servers, include the IP addresses of all DNS servers when you enter this command.

The following example shows how to configure DNS servers with the IP addresses 10.10.10.15, 10.10.10.30, 2001:DB8::1, and 2001:DB8::2 for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# dns-server value 10.10.10.15 10.10.10.30 2001:DB8::1
2001:DB8::2
ciscoasa(config-group-policy)#
```

**Step 3** If there is no default domain name specified in the **DeafultDNS DNS server group,** you must specify a default domain. Use the domain name and top level domain for example, **example.com.**

```
asa4(config)# group-policy FirstGroup attributes
asa4(config-group-policy)# default-domain value example.com
asa4(config-group-policy)#
```

**Step 4** Configure the DHCP network scope:

```
ciscoasa(config-group-policy)# dhcp-network-scope {ip_address | none}
ciscoasa(config-group-policy)#
```

DHCP scope specifies the range of IP addresses (that is, a subnetwork) that the ASA DHCP server should use to assign addresses to users of this group policy.

The following example shows how to set an IP subnetwork of 10.10.85.0 (specifying the address range of 10.10.85.0 through 10.10.85.255) for the group policy named First Group:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# dhcp-network-scope 10.10.85.0
```

## Configuring Split-Tunneling for AnyConnect Traffic

Split tunneling directs some AnyConnect network traffic through the VPN tunnel (encrypted) and other network traffic outside the VPN tunnel (unencrypted or “in the clear”).

Split tunneling is configured by creating a split tunneling policy, configuring an access control list for that policy, and adding the split tunnel policy to a group policy. When the group policy is sent to the client, that client will use the ACLs in the split tunneling policy to decide where to direct network traffic.

When you create access lists:

- You can specify both IPv4 and IPv6 addresses in an access control list.
- If you use a standard ACL, only one address or network is used.
- If you use extended ACLs, the source network is the split-tunneling network. The destination network is ignored.
- Access lists configured as any or with an address of 0.0.0.0/0.0.0.0 or ::/0 will not be sent to the client. To send all traffic over the tunnel, specify “tunnelall” when creating the split-tunnel-policy.
- Address 0.0.0.0/255.255.255.255 or ::/128 will be sent to the client only when the split-tunnel policy is **excludespecified**. This configuration tells the client not to tunnel traffic destined for any local subnets.
- AnyConnect passes traffic to all sites specified in the split tunneling policy, **and** to all sites that fall within the same subnet as the IP address assigned by the ASA. For example, if the IP address assigned by the ASA is 10.1.1.1 with a mask of 255.0.0.0, the endpoint device passes all traffic destined to 10.0.0.0/8, regardless of the split tunneling policy. Therefore, use a netmask for the assigned IP address that properly references the expected local subnet.

You can also specify a list of domains to direct split tunnel traffic. The client directs traffic to the domains in the split-dns list to the VPN, and all other traffic is in the clear.

**Prerequisites**

- You must create an access list with ACLs and ACEs.
- If you create a split tunnel policy for IPv4 networks and another for IPv6 networks, then the network list you specify in the split-tunnel-network-list command is used for both protocols. So, the network list should contain access control entries (ACEs) for both IPv4 and IPv6 traffic.

## Set the Split-Tunneling Policy

Set the rules for tunneling traffic by specifying the split-tunneling policy for IPv4 traffic:

```
ciscoasa(config-group-policy)# split-tunnel-policy {tunnelall | tunnelspecified | 
excludespecified}
ciscoasa(config-group-policy)# no split-tunnel-policy
```

Set the rules for tunneling traffic by specifying the split-tunneling policy for IPv6 traffic:

```
ciscoasa(config-group-policy)# ipv6-split-tunnel-policy {tunnelall | tunnelspecified | 
excludespecified}
```

ciscoasa(config-group-policy)# **no ipv6-split-tunnel-policy**

The policies options are:

- **tunnelspecified**— Tunnels all traffic to or from the networks specified in the Network List through the tunnel. Data to all other addresses travels in the clear and is routed by the remote user's Internet service provider.

  For versions of ASA 9.1.4 and higher, when you specify an include list, you can also specify an exclude list for a subnet inside the include range. Addresses in the excluded subnet will not be tunneled, and the rest of the include list will be. The networks in the exclusion list will not be sent over the tunnel. The exclusion list is specified using deny entries, and the inclusion list is specified using permit entries.

**Note** Networks in the exclusion list that are not a subset of the include list will be ignored by the client.

- **excludespecified** — Does not tunnel traffic to or from the networks specified in the Network List. Traffic from or to all other addresses is tunneled. The VPN client profile that is active on the client must have Local LAN Access enabled.
- **tunnelall** —Specifies that all traffic goes through the tunnel. This policy disables split tunneling. Remote users have access to the corporate network, but they do not have access to local networks. This is the default option.

**Note** Split tunneling is a traffic management feature, not a security feature. For optimum security, we recommend that you do not enable split tunneling.

The following example shows how to set a split tunneling policy of tunneling only specified networks for the group policy named FirstGroup for IPv4 and IPv6:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# split-tunnel-policy tunnelspecified

ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# ipv6-split-tunnel-policy tunnelspecified
```

## Specify a Network List for Split-Tunneling

In split tunneling, network lists determine what network traffic travels across the tunnel. AnyConnect makes split tunneling decisions on the basis of a network list, which is an ACL.

### Procedure

```
ciscoasa(config-group-policy)# split-tunnel-network-list {value access-list_name | none}
ciscoasa(config-group-policy)# no split-tunnel-network-list value [access-list_name]
```

- **value** *access-list name* — identifies an ACL that enumerates the networks to tunnel or not tunnel. The ACL can be a unified ACL with ACEs that specify both IPv4 and IPv6 addresses.
- **none** — indicates that there is no network list for split tunneling; the ASA tunnels all traffic. Specifying the **none** keyword sets a split tunneling network list with a null value, thereby disallowing split tunneling. It also prevents inheriting a default split tunneling network list from a default or specified group policy.

To delete a network list, enter the **no** form of this command. To delete all split tunneling network lists, enter the **no split-tunnel-network-list** command without arguments. This command deletes all configured network lists, including a null list if you created one by entering the **none** keyword.

When there are no split tunneling network lists, users inherit any network lists that exist in the default or specified group policy. To prevent users from inheriting such network lists, enter the **split-tunnel-network-list none** command.

### Example

The following example shows how to create a network list named FirstList, and add it to the group policy named FirstGroup. FistList is an exclusion list and an inclusion list that is a subnet of the exclusion list:

```
ciscoasa(config)# split-tunnel-policy tunnelspecified
ciscoasa(config)# access-list FirstList deny ip 10.10.10.0 255.255.255.0 any
ciscoasa(config)# access-list FirstList permit ip 10.0.0.0 255.0.0.0 any

ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# split-tunnel-network-list value FirstList
```

The following example shows how to create a network list named v6, and add the v6 split tunnel policy to the group policy named GroupPolicy_ipv6-ikev2. v6 is an exclusion list and an inclusion list that is a subnet of the exclusion list:

```
ciscoasa(config)# access-list v6 extended permit ip fd90:5000::/32 any6
ciscoasa(config)# access-list v6 extended deny ip fd90:5000:3000:2880::/64 any6

ciscoasa(config)# group-policy GroupPolicy_ipv6-ikev2 internal
ciscoasa(config)# group-policy GroupPolicy_ipv6-ikev2 attributes
ciscoasa(config-group-policy)# vpn-tunnel-protocol ikev2 ssl-client
ciscoasa(config-group-policy)# ipv6-split-tunnel-policy tunnelspecified
ciscoasa(config-group-policy)# split-tunnel-network-list value v6
```

### Verify the Split Tunnel Configuration

Run the **show runn group-policy attributes** command to verify your configuration. This example shows that the administrator has set both an IPv4 and IPv6 network policy and used the network list (unified ACL), **FirstList** for both policies.

```
ciscoasa(config-group-policy)# show runn group-policy FirstGroup attributes
group-policy FirstGroup attributes
 split-tunnel-policy tunnelspecified
 ipv6-split-tunnel-policy tunnelspecified
 split-tunnel-network-list value FirstList
```

## Configure Domain Attributes for Split Tunneling

You can specify a default domain name or a list of domains to be resolved through the split tunnel, which we refere to as split DNS.

AnyConnect 3.1 supports true split DNS functionality for Windows and Mac OS X platforms. If the group policy on the security appliance enables split-include tunneling, and if it specifies the DNS names to be tunneled, AnyConnect tunnels any DNS queries that match those names to the private DNS server. True split DNS allows tunnel access to only DNS requests that match the domains pushed to the client by the ASA. These requests are not sent in the clear. On the other hand, if the DNS requests do not match the domains pushed down by the ASA, AnyConnect lets the DNS resolver on the client operating system submit the host name in the clear for DNS resolution.

Note Split DNS supports standard and update queries (including A, AAAA, NS, TXT, MX, SOA, ANY, SRV, PTR, and CNAME). PTR queries matching any of the tunneled networks are allowed through the tunnel.

For Mac OS X, AnyConnect can use true split-DNS for a certain IP protocol only if one of the following conditions is met:

- Split-DNS is configured for one IP protocol (such as IPv4), and Client Bypass Protocol is configured for the other IP protocol (such as IPv6) in the group policy (with no address pool configured for the latter IP protocol).
- Split-DNS is configured for both IP protocols.

### Define a Default Domain Name

The ASA passes the default domain name to the AnyConnect client. The client appends the domain name to DNS queries that omit the domain field. This domain name applies only to tunneled packets.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.

```
ciscoasa(config-group-policy)# default-domain {value domain-name | none}
ciscoasa(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:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# default-domain value FirstDomain
```

### Define a List of Domains for Split Tunneling

Enter a list of domains to be resolved through the split tunnel, in addition to the default domain. 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 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:

```
ciscoasa(config-group-policy)# split-dns {value domain-name1 [domain-name2...
domain-nameN] | none}
ciscoasa(config-group-policy)# no split-dns [domain-name domain-name2 domain-nameN]
```

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:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# split-dns value Domain1 Domain2 Domain3 Domain4
```

**Note** When configuring split DNS, ensure the private DNS servers specified do not overlap with the DNS servers configured for the client platform. If they do, name resolution does not function properly and queries may be dropped.

## Configuring DHCP Intercept for Windows XP and Split Tunneling

A Microsoft XP anomaly results in the corruption of domain names if split tunnel options exceed 255 bytes. To avoid this problem, the ASA 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. The ASA 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.

```
ciscoasa(config-group-policy)# intercept-dhcp netmask {enable | disable}
ciscoasa(config-group-policy)#
```

The *netmask* variable provides the subnet mask for the tunnel IP address. The **no** form of this command removes the DHCP intercept from the configuration:

```
[no] intercept-dhcp
```

The following example shows how to set DHCP Intercepts for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# intercept-dhcp enable
```

## Setting Up a Split Exclusion Policy for Web Security

### Information about Cloud Web Security

The AnyConnect Web Security module is an endpoint component that routes HTTP traffic to a Cisco Cloud Web Security scanning proxy where Cisco Cloud Web Security evaluates it. Cisco Cloud Web Security deconstructs the elements of a Web page so that it can analyze each element simultaneously. It blocks potentially harmful content and allows benign content to come through.

With many Cisco Cloud Web Security scanning proxies spread around the world, users taking advantage of AnyConnect Web Security are able to route their traffic to the Cisco Cloud Web Security scanning proxy with the fastest response time to minimize latency.

When a user has established a VPN session, all network traffic is sent through the VPN tunnel. However, when AnyConnect users are using web security, the HTTP traffic originating at the endpoint needs to be excluded from the tunnel and sent directly to the Cloud Web Security scanning proxy.

To set up the split tunnel exclusions for traffic meant for the Cloud Web Security scanning proxy, use the **Set up split exclusion for Web Security** button in a group policy.

### Prerequisites

- You need to have access to the ASA using ASDM. This procedure cannot be performed using the command line interface.
- Web security needs to be configured for use with the AnyConnect client. See Configuring Web Security in the *AnyConnect Secure Mobility Client Administrator Guide*.
- You have created a Group Policy and assigned it a Connection Profile for AnyConnect clients configured with Web Security.

### Detailed Steps

**Step 1** Start an ASDM session for the head end you want to configure and select **Remote Access VPN > Configuration > Group Policies**.

**Step 2** Select the Group Policy you want to configure and click **Edit**.

**Step 3** Select **Advanced > Split Tunneling.**

**Step 4** Click **Set up split exclusion for Web Security.**

**Step 5** Enter a new, or select an existing, ACL used for Web Security split exclusion. ASDM will set up the ACL for use in the network list.

**Step 6** Click **Create Access List** for a new list or **Update Access List** for an existing list.

**Step 7** Click **OK**.

### What to do next

When additional scanning proxies are added, update the unified ACL you created in this procedure with new information.

## Configuring Browser Proxy Settings for use with Remote Access Clients

Follow these steps to configure the proxy server parameters for a client.

**Step 1** Configure a browser proxy server and port for a client device by entering the **msie-proxy server** command in group-policy configuration mode:

```
ciscoasa(config-group-policy)# msie-proxy server {value server[:port] | none}
ciscoasa(config-group-policy)#
```

The default value is **none**. To remove the attribute from the configuration, use the **no** form of the command.

```
ciscoasa(config-group-policy)# no msie-proxy server
ciscoasa(config-group-policy)#
```

The line containing the proxy server IP address or hostname and the port number must be less than 100 characters long.

The following example shows how to configure the IP address 192.168.10.1 as a browser proxy server, using port 880, for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# msie-proxy server value 192.168.21.1:880
ciscoasa(config-group-policy)#
```

**Step 2** Configure the browser proxy actions (“methods”) for a client device by entering the **msie-proxy method** command in group-policy configuration mode.

```
ciscoasa(config-group-policy)# msie-proxy method [auto-detect | no-modify | no-proxy |
use-server]
ciscoasa(config-group-policy)#
```

The default value is **use-server**. To remove the attribute from the configuration, use the **no** form of the command.

```
ciscoasa(config-group-policy)# no msie-proxy method [auto-detect |no-modify | no-proxy |
use-server]
ciscoasa(config-group-policy)#
```

The available methods are as follows:

- **auto-detect**—Enables the use of automatic proxy server detection in the browser for the client device.
- **no-modify**—Leaves the HTTP browser proxy server setting in the browser unchanged for this client device.
- **no-proxy—**Disables the HTTP proxy setting in the browser for the client device.
- **use-server—**Sets the HTTP proxy server setting in the browser to use the value configured in the **msie-proxy server** command.

The line containing the proxy server IP address or hostname and the port number must be less than 100 characters long.

The following example shows how to configure auto-detect as the browser proxy setting for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# msie-proxy method auto-detect
ciscoasa(config-group-policy)#
```

The following example configures the browser proxy setting for the group policy named FirstGroup to use the server QAserver, port 1001 as the server for the client device:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# msie-proxy server QAserver:port 1001
ciscoasa(config-group-policy)# msie-proxy method use-server
ciscoasa(config-group-policy)#
```

**Step 3** Configure browser proxy exception list settings for a local bypass on the client device by entering the **msie-proxy except-list** command in group-policy configuration mode. These addresses are not accessed by a proxy server. This list corresponds to the Exceptions box in the Proxy Settings dialog box.

```
ciscoasa(config-group-policy)# msie-proxy except-list {value server[:port] | none}
ciscoasa(config-group-policy)#
```

To remove the attribute from the configuration, use the **no** form of the command:

```
ciscoasa(config-group-policy)# no msie-proxy except-list
ciscoasa(config-group-policy)#
```

- **value** *server:port*—Specifies the IP address or name of an MSIE server and port that is applied for this client device. The port number is optional.
- **none**—Indicates that there is no IP address/hostname or port and prevents inheriting an exception list.

By default, msie-proxy except-list is disabled.

The line containing the proxy server IP address or hostname and the port number must be less than 100 characters long.

The following example shows how to set a browser proxy exception list, consisting of the server at IP address 192.168.20.1, using port 880, for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# msie-proxy except-list value 192.168.20.1:880
ciscoasa(config-group-policy)#
```

**Step 4** Enable or disable browser proxy local-bypass settings for a client device by entering the **msie-proxy local-bypass** command in group-policy configuration mode.

```
ciscoasa(config-group-policy)# msie-proxy local-bypass {enable | disable}
ciscoasa(config-group-policy)#
```

To remove the attribute from the configuration, use the **no** form of the command.

```
ciscoasa(config-group-policy)# no msie-proxy local-bypass {enable | disable}
ciscoasa(config-group-policy)#
```

By default, msie-proxy local-bypass is disabled.

The following example shows how to enable browser proxy local-bypass for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# msie-proxy local-bypass enable
ciscoasa(config-group-policy)#
```

# Configuring Group Policy Attributes for AnyConnect Secure Mobility Client Connections

After enabling AnyConnect client connections as described in Chapter 11, "Configuring AnyConnect VPN Client Connections", you can enable or require AnyConnect features for a group policy. Follow these steps in group-policy webvpn configuration mode:

**Step 1** Enter group policy webvpn configuration mode. For example:

```
hostname(config)# group-policy sales attributes
hostname(config-group-policy)# webvpn
```

**Step 2** To disable the permanent installation of the AnyConnect client on the endpoint computer, use the **anyconnect keep-installer** command with the **none** keyword. For example:

```
hostname(config-group-webvpn)# anyconnect keep-installer none
hostname(config-group-webvpn)#
```

The default is that permanent installation of the client is enabled. The client remains installed on the endpoint at the end of the AnyConnect session.

**Step 3** To enable compression of HTTP data over an AnyConnect SSL connection for the group policy, enter the **anyconnect ssl compression** command. By default, compression is set to **none** (disabled). To enable compression, use the **deflate** keyword. For example:

```
hostname(config-group-webvpn)# anyconnect compression deflate
hostname(config-group-webvpn)#
```

**Step 4** To enable dead peer detection (DPD) on the ASA and to set the frequency with which either the AnyConnect client or the ASA performs DPD, use the **anyconnect dpd-interval** command:

**anyconnect dpd-interval** {[**gateway** {*seconds* | **none**}] | [**client** {*seconds* | **none**}]}

By default, both the ASA and the AnyConnect client perform DPD every 30 seconds.

The gateway refers to the ASA. You can specify the frequency with which the ASA performs the DPD test as a range of from 30 to 3600 seconds (1 hour). Specifying **none** disables the DPD testing that the ASA performs. A value of 300 is recommended.

The client refers to the AnyConnect client. You can specify the frequency with which the client performs the DPD test as a range of from 30 to 3600 seconds (1 hour). Specifying **none** disables the DPD testing that the client performs. A value of 30 is recommended.

The following example configures the DPD frequency performed by the ASA (gateway) to 300 seconds, and the DPD frequency performed by the client to 30 seconds:

```
hostname(config-group-webvpn)# anyconnect dpd-interval gateway 300
hostname(config-group-webvpn)# anyconnect dpd-interval client 30
hostname(config-group-webvpn)#
```

**Step 5** You can ensure that an AnyConnect connection through a proxy, firewall, or NAT device remains open, even if the device limits the time that the connection can be idle by adjusting the frequency of keepalive messages using the **anyconnect ssl keepalive comand:**

**anyconnect ssl keepalive {none** | *seconds*}

Adjusting keepalives also ensures the AnyConnect client does not disconnect and reconnect when the remote user is not actively running a socket-based application, such as Microsoft Outlook or Microsoft Internet Explorer.

The following example configures the security appliance to enable the AnyConnect client to send keepalive messages, with a frequency of 300 seconds (5 minutes):

```
hostname(config-group-webvpn)# anyconnect ssl keepalive 300
hostname(config-group-webvpn)#
```

**Step 6** To enable the AnyConnect client to perform a re-key on an SSL session, use the **anyconnect ssl rekey** command:

**anyconnect ssl rekey** {**method** {**ssl** | **new-tunnel**} | **time** *minutes* | **none**}}

By default, re-key is disabled.

Specifying the method as new-tunnel specifies that the AnyConnect client establishes a new tunnel during SSL re-key. Specifying the method as **none** disables re-key. Specifying the method as **ssl** specifies that SSL renegotiation takes place during re-key. Instead of specifying the method, you can specify the time; that is, the number of minutes from the start of the session until the re-key takes place, from 1 through 10080 (1 week).

The following example configures the AnyConnect client to renegotiate with SSL during re-key and configures the re-key to occur 30 minutes after the session begins:

```
hostname(config-group-webvpn)# anyconnect ssl rekey method ssl
hostname(config-group-webvpn)# anyconnect ssl rekey time 30
hostname(config-group-webvpn)#
```

**Step 7** The Client Protocol Bypass feature allows you to configure how the ASA manages IPv4 traffic when it is expecting only IPv6 traffic or how it manages IPv6 traffic when it is expecting only IPv4 traffic.

When the AnyConnect client makes a VPN connection to the ASA, the ASA could assign it an IPv4, IPv6, or both an IPv4 and IPv6 address. If the ASA assigns the AnyConnect connection only an IPv4 address or only an IPv6 address, you can now configure the Client Bypass Protocol to drop network traffic for which the ASA did not assign an IP address, or allow that traffic to bypass the ASA and be sent from the client unencrypted or “in the clear”.

For example, assume that the ASA assigns only an IPv4 address to an AnyConnect connection and the endpoint is dual stacked. When the endpoint attempts to reach an IPv6 address, if Client Bypass Protocol is disabled, the IPv6 traffic is dropped; however, if Client Bypass Protocol is enabled, the IPv6 traffic is sent from the client in the clear.

Use the client-bypass-protocol command to enable or disable the client bypass protocol feature. This is the command syntax:

```
client-bypass-protocol {enable | disable}
```

The following example enables client bypass protocol:

```
hostname(config-group-policy)# client-bypass-protocol enable
hostname(config-group-policy)#
```

The following example disables client bypass protocol:

```
hostname(config-group-policy)# client-bypass-protocol disable
hostname(config-group-policy)#
```

The following example removes an enabled or disabled client bypass protocol setting:

```
hostname(config-group-policy)# no client-bypass-protocol enable
hostname(config-group-policy)#
```

**Step 8** If you have configured Load Balancing between your ASAs, specify the FQDN of the ASA in order to resolve the ASA IP address used for re-establishing the VPN session. This setting is critical to support client roaming between networks of different IP protocols (such as IPv4 to IPv6).

You cannot use the ASA FQDN present in the AnyConnect profile to derive the ASA IP address after roaming. The addresses may not match the correct device (the one the tunnel was established to) in the load balancing scenario.

If the device FQDN is not pushed to the client, the client will try to reconnect to whatever IP address the tunnel had previously established. In order to support roaming between networks of different IP protocols (from IPv4 to IPv6), AnyConnect must perform name resolution of the device FQDN after roaming, so that it can determine which ASA address to use for re-establishing the tunnel. The client uses the ASA FQDN present in its profile during the initial connection. During subsequent session reconnects, it always uses the device FQDN pushed by ASA (and configured by the administrator in the group policy), when available. If the FQDN is not configured, the ASA derives the device FQDN (and sends it to the client) from whatever is set under Device Setup > Device Name/Password and Domain Name.

If the device FQDN is not pushed by the ASA, the client cannot re-establish the VPN session after roaming between networks of different IP protocols.

Use the gateway-fqdn command to configure the FQDN of the ASA. This is the command syntax:

```
gateway-fqdn value {FQDN_Name | none}
no gateway-fqdn
```

The following example defines the FQDN of the ASA as ASAName.example.cisco.com

```
hostname(config-group-policy)# gateway-fqdn value ASAName.example.cisco.com
hostname(config-group-policy)#
```

The following example removes the FQDN of the ASA from the group policy. The group policy then inherits this value from the Default Group Policy.

```
hostname(config-group-policy)# no gateway-fqdn
hostname(config-group-policy)#
```

The following example defines the FQDN as an empty value. The global FQDN configurd using hostname and domain-name commands will be used if available.

```
hostname(config-group-policy)# gateway-fqdn none
hostname(config-group-policy)#
```

# Configuring Group Policy Attributes for IPsec (IKEv1) Clients

## Configuring Security Attributes for IPsec (IKEv1) Clients

To specify the security settings for a group, perform these steps.

**Step 1** Specify whether to let users store their login passwords on the client system, using the **password-storage** command with the **enable** keyword in group-policy configuration mode. To disable password storage, use the **password-storage** command with the **disable** keyword.

```
ciscoasa(config-group-policy)# password-storage {enable | disable}
ciscoasa(config-group-policy)#
```

For security reasons, password storage is disabled by default. Enable password storage only on systems that you know to be in secure sites.

To remove the password-storage attribute from the running configuration, enter the **no** form of this command:

```
ciscoasa(config-group-policy)# no password-storage
ciscoasa(config-group-policy)#
```

Specifying the **no** form enables inheritance of a value for password-storage from another group policy.

This command does not apply to interactive hardware client authentication or individual user authentication for hardware clients.

The following example shows how to enable password storage for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# password-storage enable
ciscoasa(config-group-policy)#
```

**Step 2** Specify whether to enable IP compression, which is disabled by default.

**Note** IP compression is not supported for IPsec IKEv2 connections.

```
ciscoasa(config-group-policy)# ip-comp {enable | disable}
ciscoasa(config-group-policy)#
```

To enable LZS IP compression, enter the **ip-comp** command with the **enable** keyword in group-policy configuration mode. To disable IP compression, enter the **ip-comp** command with the **disable** keyword.

To remove the **ip-comp** attribute from the running configuration, enter the **no** form of this command. This enables inheritance of a value from another group policy.

```
ciscoasa(config-group-policy)# no ip-comp
ciscoasa(config-group-policy)#
```

Enabling data compression might speed up data transmission rates for remote dial-in users connecting with modems.

**Caution** Data compression increases the memory requirement and CPU usage for each user session and consequently decreases the overall throughput of the ASA. For this reason, we recommend that you enable data compression only for remote users connecting with a modem. Design a group policy specific to modem users, and enable compression only for them.

**Step 3** Specify whether to require that users reauthenticate on IKE re-key by using the **re-xauth** command with the **enable** keyword in group-policy configuration mode.

**Note** IKE re-key is not supported for IKEv2 connections.

If you enable reauthentication on IKE re-key, the ASA prompts the user to enter a username and password during initial Phase 1 IKE negotiation and also prompts for user authentication whenever an IKE re-key occurs. Reauthentication provides additional security.

If the configured re-key interval is very short, users might find the repeated authorization requests inconvenient. To avoid repeated authorization requests, disable reauthentication. To check the configured re-key interval, in monitoring mode, enter the **show crypto ipsec sa** command to view the security association lifetime in seconds and lifetime in kilobytes of data. To disable user reauthentication on IKE re-key, enter the **disable** keyword. Reauthentication on IKE re-key is disabled by default.

```
ciscoasa(config-group-policy)# re-xauth {enable | disable}
ciscoasa(config-group-policy)#
```

To enable inheritance of a value for reauthentication on IKE re-key from another group policy, remove the re-xauth attribute from the running configuration by entering the **no** form of this command:

```
ciscoasa(config-group-policy)# no re-xauth
ciscoasa(config-group-policy)#
```

**Note** Reauthentication fails if there is no user at the other end of the connection.

**Step 4** Specify whether to enable perfect forward secrecy. In IPsec negotiations, perfect forward secrecy ensures that each new cryptographic key is unrelated to any previous key. A group policy can inherit a value for perfect forward secrecy from another group policy. Perfect forward secrecy is disabled by default. To enable perfect forward secrecy, use the **pfs** command with the **enable** keyword in group-policy configuration mode.

```
ciscoasa(config-group-policy)# pfs {enable | disable}
ciscoasa(config-group-policy)#
```

To disable perfect forward secrecy, enter the **pfs** command with the **disable** keyword.

To remove the perfect forward secrecy attribute from the running configuration and prevent inheriting a value, enter the **no** form of this command.

```
ciscoasa(config-group-policy)# no pfs
ciscoasa(config-group-policy)#
```

## Configuring IPsec-UDP Attributes for IKEv1 Clients

IPsec over UDP, sometimes called IPsec through NAT, lets a Cisco VPN client or hardware client connect via UDP to a ASA that is running NAT. It is disabled by default. IPsec over UDP is proprietary; it applies only to remote-access connections, and it requires mode configuration. The ASA exchanges configuration parameters with the client while negotiating SAs. Using IPsec over UDP may slightly degrade system performance.

To enable IPsec over UDP, configure the **ipsec-udp** command with the **enable** keyword in group-policy configuration mode, as follows:

```
ciscoasa(config-group-policy)# ipsec-udp {enable | disable}
ciscoasa(config-group-policy)# no ipsec-udp
```

To use IPsec over UDP, you must also configure the **ipsec-udp-port** command, as described in this section.

To disable IPsec over UDP, enter the **disable** keyword. To remove the IPsec over UDP attribute from the running configuration, enter the **no** form of this command. This enables inheritance of a value for IPsec over UDP from another group policy.

The Cisco VPN client must also be configured to use IPsec over UDP (it is configured to use it by default). The VPN 3002 requires no configuration to use IPsec over UDP.

The following example shows how to set IPsec over UDP for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# ipsec-udp enable
```

If you enabled IPsec over UDP, you must also configure the **ipsec-udp-port** command in group-policy configuration mode. This command sets a UDP port number for IPsec over UDP. In IPsec negotiations, the ASA listens on the configured port and forwards UDP traffic for that port even if other filter rules drop UDP traffic. The port numbers can range from 4001 through 49151. The default port value is 10000.

To disable the UDP port, enter the **no** form of this command. This enables inheritance of a value for the IPsec over UDP port from another group policy.

```
ciscoasa(config-group-policy)# ipsec-udp-port port
```

The following example shows how to set an IPsec UDP port to port 4025 for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# ipsec-udp-port 4025
```

## Configuring Attributes for VPN Hardware Clients

This section describes how to enable or disable secure unit authentication and user authentication or set a user authentication timeout value for VPN hardware clients. They also let you allow Cisco IP phones and LEAP packets to bypass individual user authentication and allow hardware clients using Network Extension Mode to connect.

### Configuring Secure Unit Authentication

Secure unit authentication provides additional security by requiring VPN hardware clients to authenticate with a username and password each time that the client initiates a tunnel. With this feature enabled, the hardware client does not have a saved username and password. Secure unit authentication is disabled by default.

**Note** With this feature enabled, to bring up a VPN tunnel, a user must be present to enter the username and password.

Secure unit authentication requires that you have an authentication server group configured for the connection profile the hardware client(s) use. If you require secure unit authentication on the primary ASA, be sure to configure it on any backup servers as well.

Specify whether to enable secure unit authentication by entering the **secure-unit-authentication** command with the **enable** keyword in group-policy configuration mode.

```
ciscoasa(config-group-policy)# secure-unit-authentication {enable | disable}
ciscoasa(config-group-policy)# no secure-unit-authentication
```

To disable secure unit authentication, enter the **disable** keyword. To remove the secure unit authentication attribute from the running configuration, enter the **no** form of this command. This option allows inheritance of a value for secure unit authentication from another group policy.

The following example shows how to enable secure unit authentication for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# secure-unit-authentication enable
```

### Configuring User Authentication

User authentication is disabled by default. When enabled, user authentication requires that individual users behind a hardware client authenticate to gain access to the network across the tunnel. Individual users authenticate according to the order of authentication servers that you configure.

Specify whether to enable user authentication by entering the **user-authentication** command with the **enable** keyword in group-policy configuration mode.

```
ciscoasa(config-group-policy)# user-authentication {enable | disable}
ciscoasa(config-group-policy)# no user-authentication
```

To disable user authentication, enter the **disable** keyword. To remove the user authentication attribute from the running configuration, enter the **no** form of this command. This option allows inheritance of a value for user authentication from another group policy.

If you require user authentication on the primary ASA, be sure to configure it on any backup servers as well.

The following example shows how to enable user authentication for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# user-authentication enable
```

### Configuring an Idle Timeout

Set an idle timeout for individual users behind hardware clients by entering the **user-authentication-idle-timeout** command in group-policy configuration mode. If there is no communication activity by a user behind a hardware client in the idle timeout period, the ASA terminates the client's access:

```
ciscoasa(config-group-policy)# user-authentication-idle-timeout {minutes | none}
ciscoasa(config-group-policy)# no user-authentication-idle-timeout
```

**Note** This timer terminates only the client's access through the VPN tunnel, not the VPN tunnel itself.

The idle timeout indicated in response to the **show uauth** command is always the idle timeout value of the user who authenticated the tunnel on the Cisco Easy VPN remote device.

The *minutes* parameter specifies the number of minutes in the idle timeout period. The minimum is 1 minute, the default is 30 minutes, and the maximum is 35791394 minutes.

To delete the idle timeout value, enter the **no** form of this command. This option allows inheritance of an idle timeout value from another group policy.

To prevent inheriting an idle timeout value, enter the **user-authentication-idle-timeout** command with the **none** keyword. This command sets the idle timeout with a null value, which disallows an idle timeout and prevents inheriting an user authentication idle timeout value from a default or specified group policy.

The following example shows how to set an idle timeout value of 45 minutes for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# user-authentication-idle-timeout 45
```

### Configuring IP Phone Bypass

You can allow Cisco IP phones to bypass individual user authentication behind a hardware client. To enable IP Phone Bypass, enter the **ip-phone-bypass** command with the **enable** keyword in group-policy configuration mode. IP Phone Bypass lets IP phones behind hardware clients connect without undergoing user authentication processes. IP Phone Bypass is disabled by default. If enabled, secure unit authentication remains in effect.

To disable IP Phone Bypass, enter the **disable** keyword. To remove the IP phone Bypass attribute from the running configuration, enter the **no** form of this command. This option allows inheritance of a value for IP Phone Bypass from another group policy:

```
ciscoasa(config-group-policy)# ip-phone-bypass {enable | disable}
ciscoasa(config-group-policy)# no ip-phone-bypass
```

**Note** You must configure mac-exempt to exempt the clients from authentication. Refer to the "Configuring Device Pass-Through" section on page 8-8 for more information.

### Configuring LEAP Bypass

When LEAP Bypass is enabled, LEAP packets from wireless devices behind a VPN 3002 hardware client travel across a VPN tunnel prior to user authentication. This action lets workstations using Cisco wireless access point devices establish LEAP authentication and then authenticate again per user authentication. LEAP Bypass is disabled by default.

To allow LEAP packets from Cisco wireless access points to bypass individual users authentication, enter the **leap-bypass** command with the **enable** keyword in group-policy configuration mode. To disable LEAP Bypass, enter the **disable** keyword. To remove the LEAP Bypass attribute from the running configuration, enter the **no** form of this command. This option allows inheritance of a value for LEAP Bypass from another group policy:

```
ciscoasa(config-group-policy)# leap-bypass {enable | disable}
ciscoasa(config-group-policy)# no leap-bypass
```

**Note** IEEE 802.1X is a standard for authentication on wired and wireless networks. It provides wireless LANs with strong mutual authentication between clients and authentication servers, which can provide dynamic per-user, per session wireless encryption privacy (WEP) keys, removing administrative burdens and security issues that are present with static WEP keys.

Cisco Systems has developed an 802.1X wireless authentication type called Cisco LEAP. LEAP (Lightweight Extensible Authentication Protocol) implements mutual authentication between a wireless client on one side of a connection and a RADIUS server on the other side. The credentials used for authentication, including a password, are always encrypted before they are transmitted over the wireless medium.

Cisco LEAP authenticates wireless clients to RADIUS servers. It does not include RADIUS accounting services.

This feature does not work as intended if you enable interactive hardware client authentication.

**Caution** There might be security risks to your network in allowing any unauthenticated traffic to traverse the tunnel.

The following example shows how to set LEAP Bypass for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# leap-bypass enable
```

### Enabling Network Extension Mode

Network extension mode lets hardware clients present a single, routable network to the remote private network over the VPN tunnel. IPsec encapsulates all traffic from the private network behind the hardware client to networks behind the ASA. PAT does not apply. Therefore, devices behind the ASA have direct access to devices on the private network behind the hardware client over the tunnel, and only over the tunnel, and vice versa. The hardware client must initiate the tunnel, but after the tunnel is up, either side can initiate data exchange.

Enable network extension mode for hardware clients by entering the **nem** command with the **enable** keyword in group-policy configuration mode:

```
ciscoasa(config-group-policy)# nem {enable | disable}
ciscoasa(config-group-policy)# no nem
```

To disable NEM, enter the **disable** keyword. To remove the NEM attribute from the running configuration, enter the **no** form of this command. This option allows inheritance of a value from another group policy.

The following example shows how to set NEM for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# nem enable
```

## 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 is unavailable.When you configure backup servers, the ASA 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.

Configure backup servers either on the client or on the primary ASA. If you configure backup servers on the ASA, 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:

```
ciscoasa(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.

```
ciscoasa(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 pushes a null server list.

The **keep-client-config** keyword specifies that the ASA sends no backup server information to the client. The client uses its own backup server list, if configured. This is the default.

The *server1 server 2.... server10* parameter list is a space-delimited, priority-ordered list of servers for the VPN client to use when the primary ASA is unavailable. This list identifies servers by IP address or hostname. The list can be 500 characters long, and it can contain up to10 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:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# backup-servers 10.10.10.1 192.168.10.14
```

## Configuring Network Admission Control Parameters

The group-policy NAC commands in this section all have default values. Unless you have a good reason for changing them, accept the default values for these parameters.

The ASA uses Extensible Authentication Protocol (EAP) over UDP (EAPoUDP) messaging to validate the posture of remote hosts. Posture validation involves the checking of a remote host for compliancy with safety requirements before the assignment of a network access policy. An Access Control Server must be configured for Network Admission Control before you configure NAC on the security appliance.

The Access Control Server downloads the posture token, an informational text string configurable on the ACS, to the security appliance to aid in system monitoring, reporting, debugging, and logging. A typical posture token is Healthy, Checkup, Quarantine, Infected, or Unknown. Following posture validation or clientless authentication, the ACS downloads the access policy for the session to the security appliance.

To configure Network Admission Control settings for the default group policy or an alternative group policy, perform the following steps.

**Step 1** (Optional) Configure the status query timer period. The security appliance starts the status query timer after each successful posture validation and status query response. The expiration of this timer triggers a query for changes in the host posture, referred to as a status query. Enter the number of seconds in the range 30 through 1800. The default setting is 300.

To specify the interval between each successful posture validation in a Network Admission Control session and the next query for changes in the host posture, use the **nac-sq-period** command in group-policy configuration mode:

```
ciscoasa(config-group-policy)# nac-sq-period seconds
ciscoasa(config-group-policy)#
```

To inherit the value of the status query timer from the default group policy, access the alternative group policy from which to inherit it, then use the **no** form of this command:

```
ciscoasa(config-group-policy)# no nac-sq-period [seconds]
ciscoasa(config-group-policy)#
```

The following example changes the value of the status query timer to 1800 seconds:

```
ciscoasa(config-group-policy)# nac-sq-period 1800
ciscoasa(config-group-policy)
```

The following example inherits the value of the status query timer from the default group policy:

```
ciscoasa(config-group-policy)# no nac-sq-period
ciscoasa(config-group-policy)#
```

**Step 2** (*Optional*) Configure the NAC revalidation period. The security appliance starts the revalidation timer after each successful posture validation. The expiration of this timer triggers the next unconditional posture validation. The security appliance maintains posture validation during revalidation. The default group policy becomes effective if the Access Control Server is unavailable during posture validation or revalidation. Enter the interval in seconds between each successful posture validation. The range is 300 through 86400. The default setting is 36000.

To specify the interval between each successful posture validation in a Network Admission Control session, use the **nac-reval-period** command in group-policy configuration mode:

```
ciscoasa(config-group-policy)# nac-reval-period seconds
ciscoasa(config-group-policy)#
```

To inherit the value of the Revalidation Timer from the default group policy, access the alternative group policy from which to inherit it, then use the **no** form of this command:

```
ciscoasa(config-group-policy)# no nac-reval-period [seconds]
ciscoasa(config-group-policy)#
```

The following example changes the revalidation timer to 86400 seconds:

```
ciscoasa(config-group-policy)# nac-reval-period 86400
ciscoasa(config-group-policy)
```

The following example inherits the value of the revalidation timer from the default group policy:

```
ciscoasa(config-group-policy)# no nac-reval-period
ciscoasa(config-group-policy)#
```

**Step 3** (*Optional*) Configure the default ACL for NAC. The security appliance applies the security policy associated with the selected ACL if posture validation fails. Specify **none** or an extended ACL. The default setting is **none**. If the setting is **none** and posture validation fails, the security appliance applies the default group policy.

To specify the ACL to be used as the default ACL for Network Admission Control sessions that fail posture validation, use the **nac-default-acl** command in group-policy configuration mode:

```
ciscoasa(config-group-policy)# nac-default-acl {acl-name | none}
ciscoasa(config-group-policy)#
```

To inherit the ACL from the default group policy, access the alternative group policy from which to inherit it, then use the **no** form of this command:

```
ciscoasa(config-group-policy)# no nac-default-acl [acl-name | none]
ciscoasa(config-group-policy)#
```

The elements of this command are as follows:

- *acl-name*—Specifies the name of the posture validation server group, as configured on the ASA using the **aaa-server host** command. The name must match the server-tag variable specified in that command.
- **none**—Disables inheritance of the ACL from the default group policy and does not apply an ACL to NAC sessions that fail posture validation.

Because NAC is disabled by default, VPN traffic traversing the ASA is not subject to the NAC Default ACL until NAC is enabled.

The following example identifies acl-1 as the ACL to be applied when posture validation fails:

```
ciscoasa(config-group-policy)# nac-default-acl acl-1
ciscoasa(config-group-policy)
```

The following example inherits the ACL from the default group policy:

```
ciscoasa(config-group-policy)# no nac-default-acl
ciscoasa(config-group-policy)
```

The following example disables inheritance of the ACL from the default group policy and does not apply an ACL to NAC sessions that fail posture validation:

```
ciscoasa(config-group-policy)# nac-default-acl none
ciscoasa(config-group-policy)#
```

**Step 4** Configure NAC exemptions for VPN. By default, the exemption list is empty.The default value of the filter attribute is **none**. Enter the **vpn-nac-exempt command** once for each operating system (and ACL) to be matched to exempt remote hosts from posture validation.

To add an entry to the list of remote computer types that are exempt from posture validation, use the **vpn-nac-exempt** command in group-policy configuration mode:

```
ciscoasa(config-group-policy)# vpn-nac-exempt os "os name" [filter {acl-name | none}]
[disable]
ciscoasa(config-group-policy)#
```

To disable inheritance and specify that all hosts are subject to posture validation, use the **none** keyword immediately following **vpn-nac-exempt**:

```
ciscoasa(config-group-policy)# vpn-nac-exempt none
ciscoasa(config-group-policy)#
```

To remove an entry from the exemption list, use the **no** form of this command and name the operating system (and ACL) in the entry to be removed:

```
ciscoasa(config-group-policy)# no vpn-nac-exempt [os "os name"] [filter {acl-name | none}]
[disable]
ciscoasa(config-group-policy)#
```

To remove all entries from the exemption list associated with this group policy and inherit the list from the default group policy, use the **no** form of this command without specifying additional keywords:

```
ciscoasa(config-group-policy)# no vpn-nac-exempt
ciscoasa(config-group-policy)#
```

The syntax elements for these commands are as follows:

- *acl-name*—Name of the ACL present in the ASA configuration.
- **disable**—Disables the entry in the exemption list without removing it from the list.
- **filter**—(Optional) filter to apply an ACL to filter the traffic if the computer matches the os name.
- **none**—When entered immediately after **vpn-nac-exempt**, this keyword disables inheritance and specifies that all hosts will be subject to posture validation.When entered immediately after **filter**, this keyword indicates that the entry does not specify an ACL.
- **OS—**Exempts an operating system from posture validation.
- *os name*—Operating system name. Quotation marks are required only if the name includes a space (for example, “Windows XP”).

The following example adds all hosts running Windows XP to the list of computers that are exempt from posture validation:

```
ciscoasa(config-group-policy)# vpn-nac-exempt os "Windows XP"
ciscoasa(config-group-policy)
```

The following example exempts all hosts running Windows 98 that match an ACE in the ACL named acl-1:

```
ciscoasa(config-group-policy)# vpn-nac-exempt os "Windows 98" filter acl-1
ciscoasa(config-group-policy)
```

The following example adds the same entry to the exemption list, but disables it:

```
ciscoasa(config-group-policy)# vpn-nac-exempt os "Windows 98" filter acl-1 disable
ciscoasa(config-group-policy)
```

The following example removes the same entry from the exemption list, regardless of whether it is disabled:

```
ciscoasa(config-group-policy)# no vpn-nac-exempt os "Windows 98" filter acl-1
ciscoasa(config-group-policy)
```

The following example disables inheritance and specifies that all hosts will be subject to posture validation:

```
ciscoasa(config-group-policy)# no vpn-nac-exempt none
ciscoasa(config-group-policy)
```

The following example removes all entries from the exemption list:

```
ciscoasa(config-group-policy)# no vpn-nac-exempt
ciscoasa(config-group-policy)
```

**Step 5** Enable or disable Network Admission Control by entering the following command:

```
ciscoasa(config-group-policy)# nac {enable | disable}
ciscoasa(config-group-policy)#
```

To inherit the NAC setting from the default group policy, access the alternative group policy from which to inherit it, then use the **no** form of this command:

```
ciscoasa(config-group-policy)# no nac [enable | disable]
ciscoasa(config-group-policy)#
```

By default, NAC is disabled. Enabling NAC requires posture validation for remote access. If the remote computer passes the validation checks, the ACS server downloads the access policy for the ASA to enforce. NAC is disabled by default.

An Access Control Server must be present on the network.

The following example enables NAC for the group policy:

```
ciscoasa(config-group-policy)# nac enable
ciscoasa(config-group-policy)#
```

## Configuring VPN Client Firewall Policies

A *firewall* isolates and protects a computer from the Internet by inspecting each inbound and outbound packet of data to determine whether to allow it through the firewall or to drop it. Firewalls provide extra security if remote users in a group have split tunneling configured. In this case, the firewall protects the user's computer, and thereby the corporate network, from intrusions by way of the Internet or the user's local LAN. Remote users connecting to the ASA with the VPN client can choose the appropriate firewall option.

Set personal firewall policies that the ASA pushes to the VPN client during IKE tunnel negotiation by using the **client-firewall** command in group-policy configuration mode. To delete a firewall policy, enter the **no** form of this command.

To delete all firewall policies, enter the **no client-firewall** command without arguments. This command deletes all configured firewall policies, including a null policy if you created one by entering the **client-firewall** command with the **none** keyword.

When there are no firewall policies, users inherit any that exist in the default or other group policy. To prevent users from inheriting such firewall policies, enter the **client-firewall** command with the **none** keyword.

The Add or Edit Group Policy dialog box on the Client Firewall tab, lets you configure firewall settings for VPN clients for the group policy being added or modified.

**Note** Only VPN clients running Microsoft Windows can use these firewall features. They are currently not available to hardware clients or other (non-Windows) software clients.

In the first scenario, a remote user has a personal firewall installed on the PC. The VPN client enforces firewall policy defined on the local firewall, and it monitors that firewall to make sure it is running. If the firewall stops running, the VPN client drops the connection to the ASA. (This firewall enforcement mechanism is called *Are You There (AYT)*, because the VPN client monitors the firewall by sending it periodic “are you there?” messages; if no reply comes, the VPN client knows the firewall is down and terminates its connection to the ASA.) The network administrator might configure these PC firewalls originally, but with this approach, each user can customize his or her own configuration.

In the second scenario, you might prefer to enforce a centralized firewall policy for personal firewalls on VPN client PCs. A common example would be to block Internet traffic to remote PCs in a group using split tunneling. This approach protects the PCs, and therefore the central site, from intrusions from the Internet while tunnels are established. This firewall scenario is called *push policy* or *Central Protection Policy (CPP)*. On the ASA, you create a set of traffic management rules to enforce on the VPN client, associate those rules with a filter, and designate that filter as the firewall policy. The ASA pushes this policy down to the VPN client. The VPN client then in turn passes the policy to the local firewall, which enforces it.

## Configuring AnyConnect Client Firewall Policies

Fiewall rules for the AnyConnect client can specify IPv4 and IPv6 addresses.

### Prerequisites

You have created Unified Access Rules with IPv6 addresses specified.

Table 4-1

| | Command | Description |
|---|---|---|
| **Step 1** | **webvpn**<br>**Example:**<br>hostname(config)# **group-policy ac-client-group attributes**<br>hostname(config-group-policy)# **webvpn**<br>hostname(config-group-webvpn)# | Enter webvpn group policy configuration mode. |
| **Step 2** | **anyconnect firewall-rule client-interface {private \| public} value** *[RuleName]*<br>**Example:**<br>hostname(config-group-webvpn)# **anyconnect fireall-rule client-iterface private value** *ClientFWRule* | Specifies an acces control rule for the private or public network rule. The private network rule is the rule applied to the VPN virtual dapter interface on the cient. |
| **Step 3** | **show runn group-policy** [value]<br>**Example:**<br>hostname(config-group-webvpn)# **show runn group-policy FirstGroup**<br>group-policy FirstGroup internal<br>group-policy FirstGroup attributes<br>webvpn<br>  anyconnect firewall-rule client-interface private value ClientFWRule | Displays the group policy attributes as well as the webvpn policy attribute for the group policy. |
| **Step 4** | (optional)<br>**no anyconnect firewall-rule client-ineterface private value [***RuleName***]**<br>**Example:**<br>hostname(config-group-webvpn)#**no anyconnect firewall-rule client-ineterface private value**<br>hostname(config-group-webvpn)# | Removes the client firewall rule from the private network rule. |

# Supporting a Zone Labs Integrity Server

This section introduces the Zone Labs Integrity server, also called the Check Point Integrity server, and presents an example procedure for configuring the ASA to support the Zone Labs Integrity server. The Integrity server is a central management station for configuring and enforcing security policies on remote PCs. If a remote PC does not conform to the security policy dictated by the Integrity server, it is not granted access to the private network protected by the Integrity server and ASA.

This section includes the following topics:

- Overview of the Integrity Server and ASA Interaction, page 4-74
- Configuring Integrity Server Support, page 4-74

## Overview of the Integrity Server and ASA Interaction

The VPN client software and the Integrity client software are co-resident on a remote PC. The following steps summarize the actions of the remote PC, ASA, and Integrity server in the establishment of a session between the PC and the enterprise private network:

1. The VPN client software (residing on the same remote PC as the Integrity client software) connects to the ASA and tells the ASA what type of firewall client it is.
2. After the ASA approves the client firewall type, the ASA passes Integrity server address information back to the Integrity client.
3. With the ASA acting as a proxy, the Integrity client establishes a restricted connection with the Integrity server. A restricted connection is only between the Integrity client and the Integrity server.
4. The Integrity server determines if the Integrity client is in compliance with the mandated security policies. If the Integrity client is in compliance with security policies, the Integrity server instructs the ASA to open the connection and provide the Integrity client with connection details.
5. On the remote PC, the VPN client passes connection details to the Integrity client and signals that policy enforcement should begin immediately and the Integrity client can enter the private network.
6. After the VPN connection is established, the Integrity server continues to monitor the state of the Integrity client using client heartbeat messages.

**Note** The current release of the ASA supports one Integrity server at a time, even though the user interfaces support the configuration of up to five Integrity servers. If the active Integrity server fails, configure another one on the ASA and then reestablish the VPN client session.

## Configuring Integrity Server Support

This section describes an example procedure for configuring the ASA to support the Zone Labs Integrity servers. The procedure involves configuring address, port, connection fail timeout and fail states, and SSL certificate parameters.

To configure the Integrity server, perform the following steps:

| | Command | Purpose |
|---|---|---|
| **Step 1** | **`zonelabs-Integrity server-address`** {*`hostname1`* \| *`ip-address1`*}<br>**Example:**<br>`ciscoasa(config)# zonelabs-Integrity server-address 10.0.0.5` | Configures an Integrity server using the IP address 10.0.0.5. |
| **Step 2** | **`zonelabs-integrity port`** *`port-number`*<br>**Example:**<br>`ciscoasa(config)# zonelabs-integrity port 300` | Specifies port 300 (the default port is 5054). |

| | Command | Purpose |
|---|---|---|
| **Step 3** | **zonelabs-integrity interface** *interface*<br>**Example:**<br>`ciscoasa(config)# zonelabs-integrity interface inside` | Specifies the inside interface for communications with the Integrity server. |
| **Step 4** | **zonelabs-integrity fail-timeout** *timeout*<br>**Example:**<br>`ciscoasa(config)# zonelabs-integrity fail-timeout 12` | Ensures that the ASA waits 12 seconds for a response from either the active or standby Integrity servers before declaring the Integrity server as failed and closing the VPN client connections.<br>**Note** If the connection between the ASA and the Integrity server fails, the VPN client connections remain open by default so that the enterprise VPN is not disrupted by the failure of an Integrity server. However, you may want to close the VPN connections if the Zone Labs Integrity server fails. |
| **Step 5** | **zonelabs-integrity fail-close**<br>**Example:**<br>`ciscoasa(config)# zonelabs-integrity fail-close` | Configures the ASA so that connections to VPN clients close when the connection between the ASA and the Zone Labs Integrity server fails. |
| **Step 6** | **zonelabs-integrity fail-open**<br>**Example:**<br>`ciscoasa(config)# zonelabs-integrity fail-open` | Returns the configured VPN client connection fail state to the default and ensures that the client connections remain open. |
| **Step 7** | **zonelabs-integrity ssl-certificate-port** *cert-port-number*<br>**Example:**<br>`ciscoasa(config)# zonelabs-integrity ssl-certificate-port 300` | Specifies that the Integrity server connects to port 300 (the default is port 80) on the ASA to request the server SSL certificate. |
| **Step 8** | **zonelabs-integrity ssl-client-authentication {enable \| disable}**<br>**Example:**<br>`ciscoasa(config)# zonelabs-integrity ssl-client-authentication enable` | While the server SSL certificate is always authenticated, also specifies that the client SSL certificate of the Integrity server be authenticated. |

To set the firewall client type to the Zone Labs Integrity type, enter the following command:

| Command | Purpose |
|---|---|
| **client-firewall {opt \| req} zonelabs-integrity**<br>**Example:**<br>`ciscoasa(config)# client-firewall req zonelabs-integrity` | For more information, see the "Configuring VPN Client Firewall Policies" section on page 4-71. The command arguments that specify firewall policies are not used when the firewall type is **zonelabs-integrity**, because the Integrity server determines these policies. |

## Setting Client Firewall Parameters

Enter the following commands to set the appropriate client firewall parameters. You can configure only one instance of each command. For more information, see the "Configuring VPN Client Firewall Policies" section on page 4-71.

### Cisco Integrated Firewall

```
ciscoasa(config-group-policy)# client-firewall {opt | req} cisco-integrated acl-in ACL
acl-out ACL
```

### Cisco Security Agent

```
ciscoasa(config-group-policy)# client-firewall {opt | req} cisco-security-agent
```

### No Firewall

```
ciscoasa(config-group-policy)# client-firewall none
```

### Custom Firewall

```
ciscoasa(config-group-policy)# client-firewall {opt | req} custom vendor-id num product-id
num policy {AYT | CPP acl-in ACL acl-out ACL} [description string]
```

### Zone Labs Firewalls

```
ciscoasa(config-group-policy)# client-firewall {opt | req} zonelabs-integrity
```

**Note** When the firewall type is **zonelabs-integrity**, do not include arguments. The Zone Labs Integrity Server determines the policies.

```
ciscoasa(config-group-policy)# client-firewall {opt | req} zonelabs-zonealarm policy {AYT
| CPP acl-in ACL acl-out ACL}
```

```
ciscoasa(config-group-policy)# client-firewall {opt | req} zonelabs-zonealarmorpro policy
{AYT | CPP acl-in ACL acl-out ACL}
```

```
client-firewall {opt | req} zonelabs-zonealarmpro policy {AYT | CPP acl-in ACL acl-out
ACL}
```

### Sygate Personal Firewalls

```
ciscoasa(config-group-policy)# client-firewall {opt | req} sygate-personal
```

```
ciscoasa(config-group-policy)# client-firewall {opt | req} sygate-personal-pro
```

```
ciscoasa(config-group-policy)# client-firewall {opt | req} sygate-security-agent
```

### Network Ice, Black Ice Firewall:

```
ciscoasa(config-group-policy)# client-firewall {opt | req} networkice-blackice
```

*Table 4-4 client-firewall Command Keywords and Variables*

| Parameter | Description |
|---|---|
| **acl-in** *ACL* | Provides the policy the client uses for inbound traffic. |
| **acl-out** *ACL* | Provides the policy the client uses for outbound traffic. |
| **AYT** | Specifies that the client PC firewall application controls the firewall policy. The ASA checks to make sure that the firewall is running. It asks, “Are You There?” If there is no response, the ASA tears down the tunnel. |
| **cisco-integrated** | Specifies Cisco Integrated firewall type. |
| **cisco-security-agent** | Specifies Cisco Intrusion Prevention Security Agent firewall type. |
| **CPP** | Specifies Policy Pushed as source of the VPN client firewall policy. |
| **custom** | Specifies Custom firewall type. |
| **description** *string* | Describes the firewall. |
| **networkice-blackice** | Specifies Network ICE Black ICE firewall type. |
| **none** | Indicates that there is no client firewall policy. Sets a firewall policy with a null value, thereby disallowing a firewall policy. Prevents inheriting a firewall policy from a default or specified group policy. |
| **opt** | Indicates an optional firewall type. |
| **product-id** | Identifies the firewall product. |
| **req** | Indicates a required firewall type. |
| **sygate-personal** | Specifies the Sygate Personal firewall type. |
| **sygate-personal-pro** | Specifies Sygate Personal Pro firewall type. |
| **sygate-security-agent** | Specifies Sygate Security Agent firewall type. |
| **vendor-id** | Identifies the firewall vendor. |
| **zonelabs-integrity** | Specifies Zone Labs Integrity Server firewall type. |
| **zonelabs-zonealarm** | Specifies Zone Labs Zone Alarm firewall type. |
| **zonelabs-zonealarmorpro policy** | Specifies Zone Labs Zone Alarm or Pro firewall type. |
| **zonelabs-zonealarmpro policy** | Specifies Zone Labs Zone Alarm Pro firewall type. |

The following example shows how to set a client firewall policy that requires Cisco Intrusion Prevention Security Agent for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# client-firewall req cisco-security-agent
ciscoasa(config-group-policy)#
```

## Configuring Client Access Rules

Configure rules that limit the remote access client types and versions that can connect via IPsec through the ASA by using the **client-access-rule** command in group-policy configuration mode. Construct rules according to these guidelines:

- If you do not define any rules, the ASA permits all connection types.
- When a client matches none of the rules, the ASA denies the connection. If you define a deny rule, you must also define at least one permit rule; otherwise, the ASA denies all connections.
- For both software and hardware clients, type and version must exactly match their appearance in the **show vpn-sessiondb remote** display.
- The * character is a wildcard, which you can enter multiple times in each rule. For example, **client-access rul**e **3 deny type * version 3.*** creates a priority 3 client access rule that denies all client types running versions 3.x software.
- You can construct a maximum of 25 rules per group policy.
- There is a limit of 255 characters for an entire set of rules.
- You can enter n/a for clients that do not send client type and/or version.

To delete a rule, enter the **no** form of this command. This command is equivalent to the following command:

```
ciscoasa(config-group-policy)# client-access-rule 1 deny type "Cisco VPN Client" version
4.0
```

To delete all rules, enter the **no client-access-rule command** without arguments. This deletes all configured rules, including a null rule if you created one by issuing the **client-access-rule** command with the **none** keyword.

By default, there are no access rules. When there are no client access rules, users inherit any rules that exist in the default group policy.

To prevent users from inheriting client access rules, enter the **client-access-rule** command with the **none** keyword. The result of this command is that all client types and versions can connect.

```
ciscoasa(config-group-policy)# client-access rule priority {permit | deny} type type
version {version | none}

ciscoasa(config-group-policy)# no client-access rule [priority{permit | deny} type type
version version]
```

Table 4-5 explains the meaning of the keywords and parameters in these commands.

*Table 4-5 client-access rule Command Keywords and Variables*

| Parameter | Description |
|---|---|
| **deny** | Denies connections for devices of a particular type and/or version. |
| **none** | Allows no client access rules. Sets client-access-rule to a null value, thereby allowing no restriction. Prevents inheriting a value from a default or specified group policy. |
| **permit** | Permits connections for devices of a particular type and/or version. |
| *priority* | Determines the priority of the rule. The rule with the lowest integer has the highest priority. Therefore, the rule with the lowest integer that matches a client type and/or version is the rule that applies. If a lower priority rule contradicts, the ASA ignores it. |

*Table 4-5 client-access rule Command Keywords and Variables*

| | |
|---|---|
| **type** *type* | Identifies device types via free-form strings, for example VPN 3002. A string must match exactly its appearance in the **show vpn-sessiondb remote** display, except that you can enter the * character as a wildcard. |
| **version** *version* | Identifies the device version via free-form strings, for example 7.0. A string must match exactly its appearance in the **show vpn-sessiondb remote** display, except that you can enter the * character as a wildcard. |

The following example shows how to create client access rules for the group policy named FirstGroup. These rules permit Cisco VPN clients running software version 4.x, while denying all Windows NT clients:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# client-access-rule 1 deny type WinNT version *
ciscoasa(config-group-policy)# client-access-rule 2 permit “Cisco VPN Client” version 4.*
```

**Note** The “type” field is a free-form string that allows any value, but that value must match the fixed value that the client sends to the ASA at connect time.

## Configuring Group Policy Attributes for Clientless SSL VPN Sessions

Clientless SSL VPN lets users establish a secure, remote-access VPN tunnel to the ASA using a web browser. There is no need for either a software or hardware client. Clientless SSL VPN provides easy access to a broad range of web resources and web-enabled applications from almost any computer that can reach HTTPS Internet sites. Clientless SSL VPN uses SSL and its successor, TLS1, to provide a secure connection between remote users and specific, supported internal resources that you configure at a central site. The ASA recognizes connections that need to be proxied, and the HTTP server interacts with the authentication subsystem to authenticate users. By default, clientless SSL VPN is disabled.

You can customize a configuration of clientless SSL VPN for specific internal group policies.

**Note** The webvpn mode that you enter from global configuration mode lets you configure global settings for clientless SSL VPN sessions. The webvpn mode described in this section, which you enter from group-policy configuration mode, lets you customize a configuration of group policies specifically for clientless SSL VPN sessions.

In group-policy webvpn configuration mode, you can specify whether to inherit or customize the following parameters, each of which is described in the subsequent sections:

- customizations
- html-content-filter
- homepage
- filter
- url-list
- port-forward
- port-forward-name

- sso server (single-signon server)
- auto-signon
- deny message
- AnyConnect Secure Mobility Client
- keep-alive ignore
- HTTP compression

In many instances, you define the webvpn attributes as part of configuring clientless SSL VPN, then you apply those definitions to specific groups when you configure the group-policy webvpn attributes. Enter group-policy webvpn configuration mode by using the **webvpn** command in group-policy configuration mode. Webvpn commands for group policies define access to files, URLs and TCP applications over clientless SSL VPN sessions. They also identify ACLs and types of traffic to filter. Clientless SSL VPN is disabled by default. See the description of *Chapter 14, "Introduction to Clientless SSL VPN"* for more information about configuring the attributes for clientless SSL VPN sessions.

To remove all commands entered in group-policy webvpn configuration mode, enter the **no** form of this command. These webvpn commands apply to the username or group policy from which you configure them.

```
ciscoasa(config-group-policy)# webvpn
ciscoasa(config-group-policy)# no webvpn
```

The following example shows how to enter group-policy webvpn configuration mode for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# webvpn
hostname(config-group-webvpn)#
```

## Applying Customization

Customizations determine the appearance of the windows that the user sees upon login. You configure the customization parameters as part of configuring clientless SSL VPN. To apply a previously defined web-page customization to change the look-and-feel of the web page that the user sees at login, enter the customization command in group-policy webvpn configuration mode:

```
ciscoasa(config-group-webvpn)# customization customization_name
ciscoasa(config-group-webvpn)#
```

For example, to use the customization named blueborder, enter the following command:

```
ciscoasa(config-group-webvpn)# customization blueborder
ciscoasa(config-group-webvpn)#
```

You configure the customization itself by entering the **customization** command in webvpn mode.

The following example shows a command sequence that first establishes a customization named 123 that defines a password prompt. The example then defines a group policy named testpolicy and uses the **customization** command to specify the use of the customization named 123 for clientless SSL VPN sessions:

```
ciscoasa(config)# webvpn
ciscoasa(config-webvpn)# customization 123
ciscoasa(config-webvpn-custom)# password-prompt Enter password
ciscoasa(config-webvpn)# exit
ciscoasa(config)# group-policy testpolicy nopassword
ciscoasa(config)# group-policy testpolicy attributes
ciscoasa(config-group-policy)# webvpn
ciscoasa(config-group-webvpn)# customization value 123
ciscoasa(config-group-webvpn)#
```

### Specifying a “Deny” Message

You can specify the message delivered to a remote user who logs into a clientless SSL VPN session successfully, but has no VPN privileges, by entering the **deny-message** command in group-policy webvpn configuration mode:

```
ciscoasa(config-group-webvpn)# deny-message value "message"
ciscoasa(config-group-webvpn)# no deny-message value "message"
ciscoasa(config-group-webvpn)# deny-message none
```

The **no deny-message value** command removes the message string, so that the remote user does not receive a message.

The **no deny-message none** command removes the attribute from the connection profile policy configuration. The policy inherits the attribute value.

The message can be up to 491 alphanumeric characters long, including special characters, spaces, and punctuation, but not counting the enclosing quotation marks. The text appears on the remote user’s browser upon login. When typing the string in the **deny-message value** command, continue typing even if the command wraps.

The default deny message is: “Login was successful, but because certain criteria have not been met or due to some specific group policy, you do not have permission to use any of the VPN features. Contact your IT administrator for more information.”

The first command in the following example creates an internal group policy named group2. The subsequent commands modify the attributes, including the webvpn deny message associated with that policy.

```
ciscoasa(config)# group-policy group2 internal
ciscoasa(config)# group-policy group2 attributes
ciscoasa(config-group)# webvpn
ciscoasa(config-group-webvpn)# deny-message value "Your login credentials are OK. However,
you have not been granted rights to use the VPN features. Contact your administrator for
more information."
ciscoasa(config-group-webvpn)
```

### Configuring Group Policy Filter Attributes for Clientless SSL VPN Sessions

Specify whether to filter Java, ActiveX, images, scripts, and cookies from clientless SSL VPN sessions for this group policy by using the **html-content-filter** command in webvpn mode. HTML filtering is disabled by default.

To remove a content filter, enter the **no** form of this command. To remove all content filters, including a null value created by issuing the **html-content-filter** command with the **none** keyword, enter the **no** form of this command without arguments. The **no** option allows inheritance of a value from another group policy. To prevent inheriting an html content filter, enter the **html-content-filter** command with the **none** keyword.

Using the command a second time overrides the previous setting.

```
hostname(config-group-webvpn)# html-content-filter {java | images | scripts | cookies |
none}

hostname(config-group-webvpn)# no html-content-filter [java | images | scripts | cookies |
none]
```

Table 4-6 describes the meaning of the keywords used in this command.

*Table 4-6 filter Command Keywords*

| Keyword | Meaning |
|---|---|
| **cookies** | Removes cookies from images, providing limited ad filtering and privacy. |
| **images** | Removes references to images (removes <IMG> tags). |
| **java** | Removes references to Java and ActiveX (removes <EMBED>, <APPLET>, and <OBJECT> tags). |
| **none** | Indicates that there is no filtering. Sets a null value, thereby disallowing filtering. Prevents inheriting filtering values. |
| **scripts** | Removes references to scripting (removes <SCRIPT> tags). |

The following example shows how to set filtering of JAVA and ActiveX, cookies, and images for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# webvpn
hostname(config-group-webvpn)# html-content-filter java cookies images
hostname(config-group-webvpn)#
```

### Specifying the User Home Page

Specify a URL for the web page that displays when a user in this group logs in by using the **homepage** command in group-policy webvpn configuration mode. There is no default home page.

To remove a configured home page, including a null value created by issuing the **homepage none** command, enter the **no** form of this command. The **no** option allows inheritance of a value from another group policy. To prevent inheriting a home page, enter the **homepage none** command.

The **none** keyword indicates that there is no home page for clientless SSL VPN sessions. It sets a null value, thereby disallowing a home page and prevents inheriting an home page.

The *url-string* variable following the keyword **value** provides a URL for the home page. The string must begin with either http:// or https://.

```
hostname(config-group-webvpn)# homepage {value url-string | none}
hostname(config-group-webvpn)# no homepage
hostname(config-group-webvpn)#
```

### Configuring Auto-Signon

The **auto-signon** command is a single sign-on method for users of clientless SSL VPN sessions. It passes the login credentials (username and password) to internal servers for authentication using NTLM authentication, basic authentication, or both. Multiple auto-signon commands can be entered and are processed according to the input order (early commands take precedence).

You can use the auto-signon feature in three modes: webvpn configuration, webvpn group configuration, or webvpn username configuration mode. The typical precedence behavior applies where username supersedes group, and group supersedes global. The mode you choose depends upon the desired scope of authentication.

To disable auto-signon for a particular user to a particular server, use the **no** form of the command with the original specification of IP block or URI. To disable authentication to all servers, use the **no** form without arguments. The **no** option allows inheritance of a value from the group policy.

The following example, entered in group-policy webvpn configuration mode, configures auto-signon for the user named anyuser, using basic authentication, to servers with IP addresses ranging from 10.1.1.0 to 10.1.1.255:

The following example commands configure auto-signon for users of clientless SSL VPN sessions, using either basic or NTLM authentication, to servers defined by the URI mask https://*.example.com/*:

```
ciscoasa(config)# group-policy ExamplePolicy attributes
ciscoasa(config-group-policy)# webvpn
ciscoasa(config-group-webvpn)# auto-signon allow uri https://*.example.com/* auth-type all
hostname(config-group-webvpn)#
```

The following example commands configure auto-signon for users of clientless SSL VPN sessions, using either basic or NTLM authentication, to the server with the IP address 10.1.1.0, using subnet mask 255.255.255.0:

```
ciscoasa(config)# group-policy ExamplePolicy attributes
ciscoasa(config-group-policy)# webvpn
ciscoasa(config-group-webvpn)# auto-signon allow ip 10.1.1.0 255.255.255.0 auth-type all
hostname(config-group-webvpn)#
```

### Specifying the ACL for Clientless SSL VPN Sessions

Specify the name of the ACL to use for clientless SSL VPN sessions for this group policy or username by using the **filter** command in webvpn mode. Clientless SSL VPN ACLs do not apply until you enter the **filter** command to specify them.

To remove the ACL, including a null value created by issuing the **filter none** command, enter the **no** form of this command. The **no** option allows inheritance of a value from another group policy. To prevent inheriting filter values, enter the **filter value none** command.

ACLs for clientless SSL VPN sessions do not apply until you enter the **filter** command to specify them.

You configure ACLs to permit or deny various types of traffic for this group policy. You then enter the **filter** command to apply those ACLs for clientless SSL VPN traffic.

```
hostname(config-group-webvpn)# filter {value ACLname | none}
hostname(config-group-webvpn)# no filter
```

The **none** keyword indicates that there is no **webvpntype** ACL. It sets a null value, thereby disallowing an ACL and prevents inheriting an ACL from another group policy.

The *ACLname* string following the keyword **value** provides the name of the previously configured ACL.

**Note** Clientless SSL VPN sessions do not use ACLs defined in the **vpn-filter** command.

The following example shows how to set a filter that invokes an ACL named *acl_in* for the group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# webvpn
hostname(config-group-webvpn)# filter acl_in
hostname(config-group-webvpn)#
```

### Applying a URL List

You can specify a list of URLs to appear on the clientless SSL VPN home page for a group policy. First, you must create one or more named lists by entering the **url-list** command in global configuration mode. To apply a list of servers and URLs for clientless SSL VPN sessions to a particular group policy, allowing access to the URLs in a list for a specific group policy, use the name of the list or lists you create there with the **url-list** command in group-policy webvpn configuration mode. There is no default URL list.

To remove a list, including a null value created by using the **url-list none** command**,** use the **no** form of this command. The **no** option allows inheritance of a value from another group policy. To prevent inheriting a URL list, use the **url-list none** command. Using the command a second time overrides the previous setting:

```
hostname(config-group-webvpn)# url-list {value name | none} [index]
hostname(config-group-webvpn)# no url-list
```

Table 4-7 shows the **url-list** command parameters and their meanings.

*Table 4-7 url-list Command Keywords and Variables*

| Parameter | Meaning |
|---|---|
| *index* | Indicates the display priority on the home page. |
| **none** | Sets a null value for url lists. Prevents inheriting a list from a default or specified group policy. |
| **value** *name* | Specifies the name of a previously configured list of urls. To configure such a list, use the **url-list** command in global configuration mode. |

The following example sets a URL list called FirstGroupURLs for the group policy named FirstGroup and specifies that this should be the first URL list displayed on the homepage:

```
ciscoasa(config)# group-policy FirstGroup attributes
ciscoasa(config-group-policy)# webvpn
hostname(config-group-webvpn)# url-list value FirstGroupURLs 1
hostname(config-group-webvpn)#
```

## Enabling ActiveX Relay for a Group Policy

ActiveX Relay lets a user who has established a Clientless SSL VPN session use the browser to launch Microsoft Office applications. The applications use the session to download and upload Microsoft Office documents. The ActiveX relay remains in force until the Clientless SSL VPN session closes.

To enable or disable ActiveX controls on Clientless SSL VPN sessions, enter the following command in group-policy webvpn configuration mode:

**activex-relay** {**enable | disable**}

To inherit the **activex-relay** command from the default group policy, enter the following command:

**no activex-relay**

The following commands enable ActiveX controls on clientless SSL VPN sessions associated with a given group policy:

```
ciscoasa(config-group-policy)# webvpn
ciscoasa(config-group-webvpn)# activex-relay enable
ciscoasa(config-group-webvpn)
```

## Enabling Application Access on Clientless SSL VPN Sessions for a Group Policy

To enable application access for this group policy, enter the **port-forward** command in group-policy webvpn configuration mode. Port forwarding is disabled by default.

Before you can enter the **port-forward** command in group-policy webvpn configuration mode to enable application access, you must define a list of applications that you want users to be able to use in a clientless SSL VPN session. Enter the **port-forward** command in global configuration mode to define this list.

To remove the port forwarding attribute from the group-policy configuration, including a null value created by issuing the **port-forward none** command, enter the **no** form of this command. The **no** option allows inheritance of a list from another group policy. To prevent inheriting a port forwarding list, enter the **port-forward** command with the **none** keyword. The **none** keyword indicates that there is no filtering. It sets a null value, thereby disallowing a filtering, and prevents inheriting filtering values.

The syntax of the command is as follows:

```
hostname(config-group-webvpn)# port-forward {value listname | none}
hostname(config-group-webvpn)# no port-forward
```

The *listname* string following the keyword **value** identifies the list of applications users of clientless SSL VPN sessions can access. Enter the port-forward command in webvpn configuration mode to define the list.

Using the command a second time overrides the previous setting.

The following example shows how to set a port-forwarding list called *ports1* for the internal group policy named FirstGroup:

```
ciscoasa(config)# group-policy FirstGroup internal attributes
ciscoasa(config-group-policy)# webvpn
hostname(config-group-webvpn)# port-forward value ports1
hostname(config-group-webvpn)#
```

### Configuring the Port-Forwarding Display Name

Configure the display name that identifies TCP port forwarding to end users for a particular user or group policy by using the **port-forward-name** command in group-policy webvpn configuration mode. To delete the display name, including a null value created by using the **port-forward-name none** command**,** enter the **no** form of the command. The **no** option restores the default name, Application Access. To prevent a display name, enter the **port-forward none** command. The syntax of the command is as follows:

```
hostname(config-group-webvpn)# port-forward-name {value name | none}
hostname(config-group-webvpn)# no port-forward-name
```

The following example shows how to set the name, Remote Access TCP Applications, for the internal group policy named *FirstGroup*:

```
ciscoasa(config)# group-policy FirstGroup internal attributes
ciscoasa(config-group-policy)# webvpn
hostname(config-group-webvpn)# port-forward-name value Remote Access TCP Applications
hostname(config-group-webvpn)#
```

### Configuring the Maximum Object Size to Ignore for Updating the Session Timer

Network devices exchange short keepalive messages to ensure that the virtual circuit between them is still active. The length of these messages can vary. The **keep-alive-ignore** command lets you tell the ASA to consider all messages that are less than or equal to the specified size as keepalive messages and not as traffic when updating the session timer. The range is 0 through 900 KB. The default is 4 KB.

To specify the upper limit of the HTTP/HTTPS traffic, per transaction, to ignore, use the **keep-alive-ignore** command in group-policy attributes webvpn configuration mode:

```
ciscoasa(config-group-webvpn)# keep-alive-ignore size
ciscoasa(config-group-webvpn)#
```

```
The no form of the command removes this specification from the configuration:
ciscoasa(config-group-webvpn)# no keep-alive-ignore
ciscoasa(config-group-webvpn)#
```

The following example sets the maximum size of objects to ignore as 5 KB:

```
ciscoasa(config-group-webvpn)# keep-alive-ignore 5
ciscoasa(config-group-webvpn)#
```

### Specifying HTTP Compression

Enable compression of http data over a clientless SSL VPN session for a specific group or user by entering the **http-comp** command in the group policy webvpn mode.

```
ciscoasa(config-group-webvpn)# http-comp {gzip | none}
ciscoasa(config-group-webvpn)#
```

To remove the command from the configuration and cause the value to be inherited, use the **no** form of the command:

```
ciscoasa(config-group-webvpn)# no http-comp {gzip | none}
ciscoasa(config-group-webvpn)#
```

The syntax of this command is as follows:

- **gzip—**Specifies compression is enabled for the group or user. This is the default value.
- **none—**Specifies compression is disabled for the group or user.

For clientless SSL VPN sessions, the **compression** command configured from global configuration mode overrides the **http-comp** command configured in group policy and username webvpn modes.

In the following example, compression is disabled for the group-policy sales:

```
hostname(config)# group-policy sales attributes
hostname(config-group-policy)# webvpn
hostname(config-group-webvpn)# http-comp none
hostname(config-group-webvpn)#
```

### Specifying the SSO Server

Single sign-on support, available only for clientless SSL VPN sessions, lets users access different secure services on different servers without reentering a username and password more than once. The **sso-server value** command, when entered in group-policy-webvpn mode, lets you assign an SSO server to a group policy.

To assign an SSO server to a group policy, use the **sso-server value** command in group-policy-webvpn configuration mode. This command requires that your configuration include CA SiteMinder command.

```
ciscoasa(config-group-webvpn)# sso-server value server_name
ciscoasa(config-group-webvpn)#
```

To remove the assignment and use the default policy, use the **no** form of this command. To prevent inheriting the default policy, use the **sso-server none** command.

```
ciscoasa(config-group-webvpn)# sso-server {value server_name | none}
ciscoasa(config-group-webvpn)# [no] sso-server value server_name
```

The default policy assigned to the SSO server is DfltGrpPolicy.

The following example creates the group policy “my-sso-grp-pol” and assigns it to the SSO server named “example”:

```
ciscoasa(config)# group-policy my-sso-grp-pol internal
ciscoasa(config)# group-policy my-sso-grp-pol attributes
ciscoasa(config-group-policy)# webvpn
ciscoasa(config-group-webvpn)# sso-server value example
ciscoasa(config-group-webvpn)#
```

# Configuring User Attributes

This section describes user attributes and how to configure them. It includes the following sections:

- Viewing the Username Configuration, page 4-87
- Configuring Attributes for Individual Users, page 4-87

By default, users inherit all user attributes from the assigned group policy. The ASA also lets you assign individual attributes at the user level, overriding values in the group policy that applies to that user. For example, you can specify a group policy giving all users access during business hours, but give a specific user 24-hour access.

## Viewing the Username Configuration

To display the configuration for all usernames, including default values inherited from the group policy, enter the **all** keyword with the **show running-config username** command, as follows:

```
ciscoasa# show running-config all username
ciscoasa#
```

This displays the encrypted password and the privilege level. for all users, or, if you supply a username, for that specific user. If you omit the **all** keyword, only explicitly configured values appear in this list. The following example displays the output of this command for the user named testuser:

```
ciscoasa# show running-config all username testuser
username testuser password 12RsxXQnphyr/I9Z encrypted privilege 15
```

## Configuring Attributes for Individual Users

To configure specific users, you assign a password (or no password) and attributes to a user using the **username** command, which enters username mode. Any attributes that you do not specify are inherited from the group policy.

The internal user authentication database consists of the users entered with the **username** command. The **login** command uses this database for authentication. To add a user to the ASA database, enter the **username** command in global configuration mode. To remove a user, use the **no** version of this command with the username you want to remove. To remove all usernames, use the **clear configure username** command without appending a username.

## Setting a User Password and Privilege Level

Enter the **username** command to assign a password and a privilege level for a user. You can enter the **nopassword** keyword to specify that this user does not require a password. If you do specify a password, you can specify whether that password is stored in an encrypted form.

The optional **privilege** keyword lets you set a privilege level for this user. Privilege levels range from 0 (the lowest) through 15. System administrators generally have the highest privilege level. The default level is 2.

```
ciscoasa(config)# username name {nopassword | password password [encrypted]} [privilege 
priv_level]}

ciscoasa(config)# no username [name]
```

Table 4-8 describes the meaning of the keywords and variables used in this command.

*Table 4-8 username Command Keywords and Variables*

| Keyword/Variable | Meaning |
|---|---|
| **encrypted** | Indicates that the password is encrypted. |
| *name* | Provides the name of the user. |
| **nopassword** | Indicates that this user needs no password. |
| **password** *password* | Indicates that this user has a password, and provides the password. |
| **privilege** *priv_level* | Sets a privilege level for this user. The range is from 0 to 15, with lower numbers having less ability to use commands and administer the ASA. The default privilege level is 2. The typical privilege level for a system administrator is 15. |

By default, VPN users that you add with this command have no attributes or group policy association. You must explicitly configure all values.

The following example shows how to configure a user named anyuser with an encrypted password of pw_12345678 and a privilege level of 12:

```
ciscoasa(config)# username anyuser password pw_12345678 encrypted privilege 12
ciscoasa(config)#
```

## Configuring User Attributes

After configuring the user’s password (if any) and privilege level, you set the other attributes. These can be in any order. To remove any attribute-value pair, enter the **no** form of the command.

Enter username mode by entering the **username** command with the **attributes** keyword:

```
ciscoasa(config)# username name attributes
ciscoasa(config-username)#
```

The prompt changes to indicate the new mode. You can now configure the attributes.

## Configuring VPN User Attributes

The VPN user attributes set values specific to VPN connections, as described in the following sections.

### Configuring Inheritance

You can let users inherit from the group policy the values of attributes that you have not configured at the username level. To specify the name of the group policy from which this user inherits attributes, enter the **vpn-group-policy** command. By default, VPN users have no group-policy association:

```
ciscoasa(config-username)# vpn-group-policy group-policy-name
ciscoasa(config-username)# no vpn-group-policy group-policy-name
```

For an attribute that is available in username mode, you can override the value of an attribute in a group policy for a particular user by configuring it in username mode.

The following example shows how to configure a user named anyuser to use attributes from the group policy named FirstGroup:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# vpn-group-policy FirstGroup
ciscoasa(config-username)#
```

### Configuring Access Hours

Associate the hours that this user is allowed to access the system by specifying the name of a configured time-range policy:

To remove the attribute from the running configuration, enter the **no** form of this command. This option allows inheritance of a time-range value from another group policy. To prevent inheriting a value, enter the **vpn-access-hours none** command. The default is unrestricted access.

```
ciscoasa(config-username)# vpn-access-hours value {time-range | none}
ciscoasa(config-username)# vpn-access-hours value none
ciscoasa(config)#
```

The following example shows how to associate the user named anyuser with a time-range policy called 824:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# vpn-access-hours 824
ciscoasa(config-username)#
```

### Configuring Maximum Simultaneous Logins

Specify the maximum number of simultaneous logins allowed for this user. The range is 0 through 2147483647. The default is 3 simultaneous logins. To remove the attribute from the running configuration, enter the **no** form of this command. Enter 0 to disable login and prevent user access.

```
ciscoasa(config-username)# vpn-simultaneous-logins integer
ciscoasa(config-username)# no vpn-simultaneous-logins
ciscoasa(config-username)# vpn-session-timeout alert-interval none
```

**Note** While the maximum limit for the number of simultaneous logins is very large, allowing several could compromise security and affect performance.

The following example shows how to allow a maximum of 4 simultaneous logins for the user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# vpn-simultaneous-logins 4
ciscoasa(config-username)#
```

### Configuring the Idle Timeout

Specify the idle timeout period in minutes, or enter **none** to disable the idle timeout. If there is no communication activity on the connection in this period, the ASA terminates the connection. You can optionally set the alert interval, or leave the default of one minute.

The range is 1 through 35791394 minutes. The default is 30 minutes. To allow an unlimited timeout period, and thus prevent inheriting a timeout value, enter the **vpn-idle-timeout** command with the **none** keyword. To remove the attribute from the running configuration, enter the **no** form of this command.

```
ciscoasa(config-username)# vpn-idle-timeout {minutes | none} alert-interval {minutes}
ciscoasa(config-username)# no vpn-idle-timeout alert-interval
ciscoasa(config-username)# vpn-idle-timeout alert-interval none
```

The following example shows how to set a VPN idle timeout of 15 minutes and alert interval of 3 minutes for the user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# vpn-idle-timeout 30 alert-interval 3
ciscoasa(config-username)#
```

### Configuring the Maximum Connect Time

Specify the maximum user connection time in minutes, or enter **none** to allow unlimited connection time and prevent inheriting a value for this attribute. At the end of this period of time, the ASA terminates the connection. You can optionally set the alert interval, or leave the default of one minute.

The range is 1 through 35791394 minutes. There is no default timeout. To allow an unlimited timeout period, and thus prevent inheriting a timeout value, enter the **vpn-session-timeout** command with the **none** keyword. To remove the attribute from the running configuration, enter the **no** form of this command.

```
ciscoasa(config-username)# vpn-session-timeout {minutes | none} alert-interval {minutes}
ciscoasa(config-username)# no vpn-session-timeout alert-interval
ciscoasa(config-username)#
```

The following example shows how to set a VPN session timeout of 180 minutes for the user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# vpn-session-timeout 180 alert-interval {minutes}
ciscoasa(config-username)#
```

### Applying an ACL Filter

Specify the name of a previously-configured, user-specific ACL to use as a filter for VPN connections. To disallow an ACL and prevent inheriting an ACL from the group policy, enter the **vpn-filter** command with the none keyword. To remove the ACL, including a null value created by issuing the **vpn-filter none** command, enter the **no** form of this command. The **no** option allows inheritance of a value from the group policy. There are no default behaviors or values for this command.

You configure ACLs to permit or deny various types of traffic for this user. You then use the **vpn-filter** command to apply those ACLs.

```
ciscoasa(config-username)# vpn-filter {value ACL_name | none}
ciscoasa(config-username)# no vpn-filter
ciscoasa(config-username)#
```

**Note** Clientless SSL VPN does not use ACLs defined in the **vpn-filter** command.

The following example shows how to set a filter that invokes an ACL named acl_vpn for the user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# vpn-filter value acl_vpn
ciscoasa(config-username)#
```

### Specifying the IPv4 Address and Netmask

Specify the IP address and netmask to assign to a particular user. To remove the IP address, enter the **no** form of this command.

```
ciscoasa(config-username)# vpn-framed-ip-address {ip_address}
ciscoasa(config-username)# no vpn-framed-ip-address
ciscoasa(config-username)
```

The following example shows how to set an IP address of 10.92.166.7 for a user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# vpn-framed-ip-address 10.92.166.7
ciscoasa(config-username)
```

Specify the network mask to use with the IP address specified in the previous step. If you used the **no vpn-framed-ip-address** command, do not specify a network mask. To remove the subnet mask, enter the **no** form of this command. There is no default behavior or value.

```
ciscoasa(config-username)# vpn-framed-ip-netmask {netmask}
ciscoasa(config-username)# no vpn-framed-ip-netmask
ciscoasa(config-username)
```

The following example shows how to set a subnet mask of 255.255.255. 254 for a user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# vpn-framed-ip-netmask 255.255.255.254
ciscoasa(config-username)
```

### Specifying the IPv6 Address and Netmask

Specify the IPv6 address and netmask to assign to a particular user. To remove the IP address, enter the **no** form of this command.

```
ciscoasa(config-username)# vpn-framed-ipv6-address {ip_address}
ciscoasa(config-username)# no vpn-framed-ipv6-address
ciscoasa(config-username)
```

The following example shows how to set an IP address and netmask of 2001::3000:1000:2000:1/64 for a user named anyuser. This address indicates a prefix value of 2001:0000:0000:0000 and an interface ID of 3000:1000:2000:1.

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# vpn-framed-ipv6-address 2001::3000:1000:2000:1/64
ciscoasa(config-username)
```

### Specifying the Tunnel Protocol

Specify the VPN tunnel types (IPsec or clientless SSL VPN) that this user can use. The default is taken from the default group policy, the default for which is IPsec. To remove the attribute from the running configuration, enter the **no** form of this command.

```
ciscoasa(config-username)# vpn-tunnel-protocol {webvpn | IPsec}
ciscoasa(config-username)# no vpn-tunnel-protocol [webvpn | IPsec]
ciscoasa(config-username)
```

The parameter values for this command are as follows:

- **IPsec—**Negotiates an IPsec tunnel between two peers (a remote access client or another secure gateway). Creates security associations that govern authentication, encryption, encapsulation, and key management.
- **webvpn—**Provides clientless SSL VPN access to remote users via an HTTPS-enabled web browser, and does not require a client

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 clientless SSL VPN and IPsec tunneling modes for the user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# vpn-tunnel-protocol webvpn
ciscoasa(config-username)# vpn-tunnel-protocol IPsec
ciscoasa(config-username)
```

### Restricting Remote User Access

Configure the **group-lock** attribute with the **value** keyword to restrict remote users to access only through the specified, preexisting connection profile. Group-lock restricts users by checking whether the group configured in the VPN client is the same as the connection profile to which the user is assigned. If it is not, the ASA prevents the user from connecting. If you do not configure group-lock, the ASA authenticates users without regard to the assigned group.

To remove the **group-lock** attribute from the running configuration, enter the **no** form of this command. This option allows inheritance of a value from the group policy. To disable group-lock, and to prevent inheriting a group-lock value from a default or specified group policy, enter the **group-lock** command with the **none** keyword.

```
ciscoasa(config-username)# group-lock {value tunnel-grp-name | none}
ciscoasa(config-username)# no group-lock
ciscoasa(config-username)
```

The following example shows how to set group lock for the user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# group-lock value tunnel-group-name
ciscoasa(config-username)
```

### Enabling Password Storage for Software Client Users

Specify whether to let users store their login passwords on the client system. Password storage is disabled by default. Enable password storage only on systems that you know to be in secure sites. To disable password storage, enter the **password-storage** command with the **disable** keyword. To remove the password-storage attribute from the running configuration, enter the **no** form of this command. This enables inheritance of a value for password-storage from the group policy.

```
ciscoasa(config-username)# password-storage {enable | disable}
ciscoasa(config-username)# no password-storage
ciscoasa(config-username)
```

This command has no bearing on interactive hardware client authentication or individual user authentication for hardware clients.

The following example shows how to enable password storage for the user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# password-storage enable
ciscoasa(config-username)
```

## Configuring Clientless SSL VPN Access for Specific Users

The following sections describe how to customize a configuration for specific users of clientless SSL VPN sessions. Enter username webvpn configuration mode by using the **webvpn** command in username configuration mode. Clientless SSL VPN lets users establish a secure, remote-access VPN tunnel to the ASA using a web browser. There is no need for either a software or hardware client. Clientless SSL VPN provides easy access to a broad range of web resources and web-enabled applications from almost any computer that can reach HTTPS Internet sites. Clientless SSL VPN uses SSL and its successor, TLS1, to provide a secure connection between remote users and specific, supported internal resources that you configure at a central site. The ASA recognizes connections that need to be proxied, and the HTTP server interacts with the authentication subsystem to authenticate users.

The username webvpn configuration mode commands define access to files, URLs and TCP applications over clientless SSL VPN sessions. They also identify ACLs and types of traffic to filter. Clientless SSL VPN is disabled by default. These **webvpn** commands apply only to the username from which you configure them. Notice that the prompt changes, indicating that you are now in username webvpn configuration mode.

```
ciscoasa(config-username)# webvpn
ciscoasa(config-username-webvpn)#
```

To remove all commands entered in username webvpn configuration mode, use the **no** form of this command:

```
ciscoasa(config-username)# no webvpn
ciscoasa(config-username)#
```

You do not need to configure clientless SSL VPN to use e-mail proxies.

**Note** The webvpn mode that you enter from global configuration mode lets you configure global settings for clientless SSL VPN sessions. The username webvpn configuration mode described in this section, which you enter from username mode, lets you customize the configuration of specific users specifically for clientless SSL VPN sessions.

In username webvpn configuration mode, you can customize the following parameters, each of which is described in the subsequent steps:

- customizations
- deny message
- html-content-filter
- homepage
- filter
- url-list
- port-forward
- port-forward-name
- sso server (single-signon server)
- auto-signon
- AnyConnect Secure Mobility Client
- keep-alive ignore
- HTTP compression

The following example shows how to enter username webvpn configuration mode for the username anyuser attributes:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# webvpn
ciscoasa(config-username-webvpn)#
```

### Specifying the Content/Objects to Filter from the HTML

To filter Java, ActiveX, images, scripts, and cookies for clientless SSL VPN sessions for this user, enter the **html-content-filter** command in username webvpn configuration mode. To remove a content filter, enter the **no** form of this command. To remove all content filters, including a null value created by issuing the **html-content-filter none** command, enter the **no** form of this command without arguments. The **no** option allows inheritance of a value from the group policy. To prevent inheriting an HTML content filter, enter the **html-content-filter none** command. HTML filtering is disabled by default.

Using the command a second time overrides the previous setting.

```
hostname(config-username-webvpn)# html-content-filter {java | images | scripts | cookies |
none}

hostname(config-username-webvpn)# no html-content-filter [java | images | scripts |
cookies | none]
```

The keywords used in this command are as follows:

- **cookies—**Removes cookies from images, providing limited ad filtering and privacy.
- **images—**Removes references to images (removes <IMG> tags).
- **java—**Removes references to Java and ActiveX (removes <EMBED>, <APPLET>, and <OBJECT> tags.
- **none—**Indicates that there is no filtering. Sets a null value, thereby disallowing filtering. Prevents inheriting filtering values.
- **scripts**—Removes references to scripting (removes <SCRIPT> tags).

The following example shows how to set filtering of JAVA and ActiveX, cookies, and images for the user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# webvpn
hostname(config-username-webvpn)# html-content-filter java cookies images
ciscoasa(config-username-webvpn)#
```

### Specifying the User Home Page

To specify a URL for the web page that displays when this user logs into clientless SSL VPN session, enter the **homepage** command in username webvpn configuration mode. To remove a configured home page, including a null value created by issuing the **homepage none** command, enter the **no** form of this command. The **no** option allows inheritance of a value from the group policy. To prevent inheriting a home page, enter the **homepage none** command.

The **none** keyword indicates that there is no clientless SSL VPN home page. It sets a null value, thereby disallowing a home page and prevents inheriting a home page.

The *url-string* variable following the keyword **value** provides a URL for the home page. The string must begin with either http:// or https://.

There is no default home page.

```
hostname(config-username-webvpn)# homepage {value url-string | none}
hostname(config-username-webvpn)# no homepage
ciscoasa(config-username-webvpn)#
```

The following example shows how to specify www.example.com as the home page for the user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# webvpn
hostname(config-username-webvpn)# homepage value www.example.com
ciscoasa(config-username-webvpn)#
```

### Applying Customization

Customizations determine the appearance of the windows that the user sees upon login. You configure the customization parameters as part of configuring clientless SSL VPN. To apply a previously defined web-page customization to change the look-and-feel of the web page that the user sees at login, enter the customization command in username webvpn configuration mode:

```
ciscoasa(config-username-webvpn)# customization {none | value customization_name}
ciscoasa(config-username-webvpn)#
```

For example, to use the customization named blueborder, enter the following command:

```
ciscoasa(config-username-webvpn)# customization value blueborder
ciscoasa(config-username-webvpn)#
```

You configure the customization itself by entering the **customization** command in webvpn mode.

The following example shows a command sequence that first establishes a customization named 123 that defines a password prompt. The example then defines a tunnel-group named test and uses the **customization** command to specify the use of the customization named 123:

```
ciscoasa(config)# webvpn
ciscoasa(config-webvpn)# customization 123
ciscoasa(config-webvpn-custom)# password-prompt Enter password
ciscoasa(config-webvpn)# exit
ciscoasa(config)# username testuser nopassword
ciscoasa(config)# username testuser attributes
ciscoasa(config-username-webvpn)# webvpn
ciscoasa(config-username-webvpn)# customization value 123
ciscoasa(config-username-webvpn)#
```

### Specifying a "Deny" Message

You can specify the message delivered to a remote user who logs into clientless SSL VPN session successfully, but has no VPN privileges by entering the **deny-message** command in username webvpn configuration mode:

```
ciscoasa(config-username-webvpn)# deny-message value "message"
ciscoasa(config-username-webvpn)# no deny-message value "message"
ciscoasa(config-username-webvpn)# deny-message none
```

The **no deny-message value** command removes the message string, so that the remote user does not receive a message.

The **no deny-message none** command removes the attribute from the connection profile policy configuration. The policy inherits the attribute value.

The message can be up to 491 alphanumeric characters long, including special characters, spaces, and punctuation, but not counting the enclosing quotation marks. The text appears on the remote user's browser upon login. When typing the string in the **deny-message value** command, continue typing even if the command wraps.

The default deny message is: "Login was successful, but because certain criteria have not been met or due to some specific group policy, you do not have permission to use any of the VPN features. Contact your IT administrator for more information."

The first command in the following example enters username mode and configures the attributes for the user named anyuser. The subsequent commands enter username webvpn configuration mode and modify the deny message associated with that user.

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# webvpn
ciscoasa(config-username-webvpn)# deny-message value "Your login credentials are OK.
However, you have not been granted rights to use the VPN features. Contact your
administrator for more information."
ciscoasa(config-username-webvpn)
```

### Specifying the ACL for Clientless SSL VPN Sessions

To specify the name of the ACL to use for clientless SSL VPN sessions for this user, enter the **filter** command in username webvpn configuration mode. To remove the ACL, including a null value created by issuing the **filter none** command, enter the **no** form of this command. The **no** option allows inheritance of a value from the group policy. To prevent inheriting filter values, enter the **filter value none** command.

Clientless SSL VPN ACLs do not apply until you enter the **filter** command to specify them.

You configure ACLs to permit or deny various types of traffic for this user. You then enter the **filter** command to apply those ACLs for clientless SSL VPN traffic.

```
hostname(config-username-webvpn)# filter {value ACLname | none}
hostname(config-username-webvpn)# no filter
ciscoasa(config-username-webvpn)#
```

The **none** keyword indicates that there is no **webvpntype** ACL. It sets a null value, thereby disallowing an ACL and prevents inheriting an ACL from another group policy.

The *ACLname* string following the keyword **value** provides the name of the previously configured ACL.

**Note** Clientless SSL VPN does not use ACLs defined in the **vpn-filter** command.

The following example shows how to set a filter that invokes an ACL named *acl_in* for the user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# webvpn
hostname(config-username-webvpn)# filter acl_in
ciscoasa(config-username-webvpn)#
```

### Applying a URL List

You can specify a list of URLs to appear on the home page for a user who has established a clientless SSL VPN session. First, you must create one or more named lists by entering the **url-list** command in global configuration mode. To apply a list of servers and URLs to a particular user of clientless SSL VPN, enter the **url-list** command in username webvpn configuration mode.

To remove a list, including a null value created by using the **url-list none** command**,** enter the **no** form of this command. The **no** option allows inheritance of a value from the group policy. To prevent inheriting a url list, enter the **url-list none** command.

```
hostname(config-username-webvpn)# url-list {listname displayname url | none}
hostname(config-username-webvpn)# no url-list
```

The keywords and variables used in this command are as follows:

- *displayname*—Specifies a name for the URL. This name appears on the portal page in the clientless SSL VPN session.
- *listname*—Identifies a name by which to group URLs.
- **none**—Indicates that there is no list of URLs. Sets a null value, thereby disallowing a URL list. Prevents inheriting URL list values.
- *url*—Specifies a URL that users of clientless SSL VPN can access.

There is no default URL list.

Using the command a second time overrides the previous setting.

The following example shows how to set a URL list called AnyuserURLs for the user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# webvpn
hostname(config-username-webvpn)# url-list value AnyuserURLs
ciscoasa(config-username-webvpn)#
```

### Enabling ActiveX Relay for a User

ActiveX Relay lets a user who has established a Clientless SSL VPN session use the browser to launch Microsoft Office applications. The applications use the session to download and upload Microsoft Office documents. The ActiveX relay remains in force until the Clientless SSL VPN session closes.

To enable or disable ActiveX controls on Clientless SSL VPN sessions, enter the following command in username webvpn configuration mode:

**activex-relay** {**enable** | **disable**}

To inherit the **activex-relay** command from the group policy, enter the following command:

**no activex-relay**

The following commands enable ActiveX controls on Clientless SSL VPN sessions associated with a given username:

```
ciscoasa(config-username-policy)# webvpn
ciscoasa(config-username-webvpn)# activex-relay enable
ciscoasa(config-username-webvpn)
```

### Enabling Application Access for Clientless SSL VPN Sessions

To enable application access for this user, enter the **port-forward** command in username webvpn configuration mode. Port forwarding is disabled by default.

To remove the port forwarding attribute from the configuration, including a null value created by issuing the **port-forward none** command, enter the **no** form of this command. The **no** option allows inheritance of a list from the group policy. To disallow filtering and prevent inheriting a port forwarding list, enter the **port-forward** command with the **none** keyword.

```
hostname(config-username-webvpn)# port-forward {value listname | none}
hostname(config-username-webvpn)# no port-forward
ciscoasa(config-username-webvpn)#
```

The *listname* string following the keyword **value** identifies the list of applications users of clientless SSL VPN can access. Enter the **port-forward** command in configuration mode to define the list.

Using the command a second time overrides the previous setting.

Before you can enter the **port-forward** command in username webvpn configuration mode to enable application access, you must define a list of applications that you want users to be able to use in a clientless SSL VPN session. Enter the **port-forward** command in global configuration mode to define this list.

The following example shows how to configure a portforwarding list called ports1:

```
ciscoasa(config-group-policy)# webvpn
hostname(config-username-webvpn)# port-forward value ports1
ciscoasa(config-username-webvpn)#
```

### Configuring the Port-Forwarding Display Name

Configure the display name that identifies TCP port forwarding to end users for a particular user by using the **port-forward-name** command in username webvpn configuration mode. To delete the display name, including a null value created by using the **port-forward-name none** command**,** enter the **no** form of the command. The **no** option restores the default name, Application Access. To prevent a display name, enter the **port-forward none** command.

```
hostname(config-username-webvpn)# port-forward-name {value name | none}
hostname(config-username-webvpn)# no port-forward-name
```

The following example shows how to configure the port-forward name test:

```
ciscoasa(config-group-policy)# webvpn
hostname(config-username-webvpn)# port-forward-name value test
ciscoasa(config-username-webvpn)#
```

### Configuring the Maximum Object Size to Ignore for Updating the Session Timer

Network devices exchange short keepalive messages to ensure that the virtual circuit between them is still active. The length of these messages can vary. The **keep-alive-ignore** command lets you tell the ASA to consider all messages that are less than or equal to the specified size as keepalive messages and not as traffic when updating the session timer. The range is 0 through 900 KB. The default is 4 KB.

To specify the upper limit of the HTTP/HTTPS traffic, per transaction, to ignore, use the **keep-alive-ignore** command in group-policy attributes webvpn configuration mode:

```
ciscoasa(config-group-webvpn)# keep-alive-ignore size
ciscoasa(config-group-webvpn)#
```

```
The no form of the command removes this specification from the configuration:
ciscoasa(config-group-webvpn)# no keep-alive-ignore
ciscoasa(config-group-webvpn)#
```

The following example sets the maximum size of objects to ignore as 5 KB:

```
ciscoasa(config-group-webvpn)# keep-alive-ignore 5
ciscoasa(config-group-webvpn)#
```

### Configuring Auto-Signon

To automatically submit the login credentials of a particular user of clientless SSL VPN to internal servers using NTLM, basic HTTP authentication or both, use the **auto-signon** command in username webvpn configuration mode.

The **auto-signon** command is a single sign-on method for users of clientless SSL VPN sessions. It passes the login credentials (username and password) to internal servers for authentication using NTLM authentication, basic authentication, or both. Multiple auto-signon commands can be entered and are processed according to the input order (early commands take precedence).

You can use the auto-signon feature in three modes: webvpn configuration, webvpn group configuration, or webvpn username configuration mode. The typical precedence behavior applies where username supersedes group, and group supersedes global. The mode you choose will depend upon the desired scope of authentication.

To disable auto-signon for a particular user to a particular server, use the **no** form of the command with the original specification of IP block or URI. To disable authentication to all servers, use the **no** form without arguments. The **no** option allows inheritance of a value from the group policy.

The following example commands configure auto-signon for a user of clientless SSL VPN named anyuser, using either basic or NTLM authentication, to servers defined by the URI mask https://*.example.com/*:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# webvpn
ciscoasa(config-username-webvpn)# auto-signon allow uri https://*.example.com/* auth-type
all
```

```
The following example commands configure auto-signon for a user of clientless SSL VPN named
anyuser, using either basic or NTLM authentication, to the server with the IP address
10.1.1.0, using subnet mask 255.255.255.0:
```

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# webvpn
ciscoasa(config-username-webvpn)# auto-signon allow ip 10.1.1.0 255.255.255.0 auth-type
all
hostname(config-username-webvpn)#
```

### Specifying HTTP Compression

Enable compression of http data over a clientless SSL VPN session for a specific user by entering the **http-comp** command in the username webvpn configuration mode.

```
ciscoasa(config-username-webvpn)# http-comp {gzip | none}
ciscoasa(config-username-webvpn)#
```

To remove the command from the configuration and cause the value to be inherited, use the **no** form of the command:

```
ciscoasa(config-username-webvpn)# no http-comp {gzip | none}
ciscoasa(config-username-webvpn)#
```

The syntax of this command is as follows:

- **gzip—**Specifies compression is enabled for the group or user. This is the default value.
- **none—**Specifies compression is disabled for the group or user.

For clientless SSL VPN session, the **compression** command configured from global configuration mode overrides the **http-comp** command configured in group policy and username webvpn modes.

In the following example, compression is disabled for the username testuser:

```
hostname(config)# username testuser internal
hostname(config)# username testuser attributes
hostname(config-username)# webvpn
hostname(config-username-webvpn)# http-comp none
hostname(config-username-webvpn)#
```

### Specifying the SSO Server

Single sign-on support, available only for clientless SSL VPN sessions, lets users access different secure services on different servers without reentering a username and password more than once. The **sso-server value** command, when entered in username-webvpn mode, lets you assign an SSO server to a user.

To assign an SSO server to a user, use the **sso-server value** command in username-webvpn configuration mode. This command requires that your configuration include CA SiteMinder command.

```
ciscoasa(config-username-webvpn)# sso-server value server_name
ciscoasa(config-username-webvpn)#
```

To remove the assignment and use the default policy, use the **no** form of this command. To prevent inheriting the default policy, use the **sso-server none** command.

```
ciscoasa(config-username-webvpn)# sso-server {value server_name | none}
ciscoasa(config-username-webvpn)# [no] sso-server value server_name
```

The default policy assigned to the SSO server is DfltGrpPolicy.

The following example assigns the SSO server named example to the user named anyuser:

```
ciscoasa(config)# username anyuser attributes
ciscoasa(config-username)# webvpn
ciscoasa(config-username-webvpn)# sso-server value example
ciscoasa(config-username-webvpn)#
```