



CHAPTER 30

Configuring Tunnel Groups, Group Policies, and Users

This chapter describes how to configure VPN tunnel groups, group policies, and users. This chapter includes the following sections.

- [Overview of Tunnel Groups, Group Policies, and Users, page 30-1](#)
- [Configuring Tunnel Groups, page 30-5](#)
- [Group Policies, page 30-31](#)
- [Configuring User Attributes, page 30-72](#)

In summary, you first configure tunnel groups 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 Tunnel Groups, Group Policies, and Users

Groups and users are core concepts in managing the security of virtual private networks (VPNs) and in configuring the security appliance. 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*. *Tunnel groups* identify 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.

Tunnel groups and group policies simplify system management. To streamline the configuration task, the security appliance provides a default LAN-to-LAN tunnel group, a default remote access tunnel group, a default WebVPN tunnel group, and a default group policy (DfltGrpPolicy). The default tunnel groups 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 tunnel groups 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. Tunnel groups and group policies provide the flexibility to do so securely.

**Note**

The security appliance 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 tunnel groups. For more information about using object groups, see [Chapter 16, “Identifying Traffic with Access Lists.”](#)

Tunnel Groups

A tunnel group 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. Tunnel groups include a small number of attributes that pertain to creating the tunnel itself. Tunnel groups include a pointer to a group policy that defines user-oriented attributes.

The security appliance provides the following default tunnel groups: DefaultL2Lgroup for LAN-to-LAN connections, DefaultRAGroup for remote access connections, and DefaultWEBVPNGroup for WebVPN connections. You can modify these default tunnel groups, but you cannot delete them. You can also create one or more tunnel groups specific to your environment. Tunnel groups are local to the security appliance and are not configurable on external servers.

Tunnel groups specify the following attributes:

- [General Tunnel-Group Connection Parameters, page 30-2](#)
- [IPSec Tunnel-Group Connection Parameters, page 30-3](#)
- [WebVPN Tunnel-Group Connection Parameters, page 30-4](#)

General Tunnel-Group Connection Parameters

General parameters are common to both IPSec and WebVPN connections. The general parameters include the following:

- Tunnel group name—You specify a tunnel-group name when you add or edit a tunnel group. The following considerations apply:
 - For clients that use preshared keys to authenticate, the tunnel group name is the same as the group name that an IPSec client passes to the security appliance.
 - Clients that use certificates to authenticate pass this name as part of the certificate, and the security appliance extracts the name from the certificate.
- Connection type—Connection types include IPSec remote access, IPSec LAN-to-LAN, and WebVPN. A tunnel group can have only one connection type.
- Authentication, Authorization, and Accounting servers—These parameters identify the server groups or lists that the security appliance 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 security appliance 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 security appliance 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 security appliance 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).

When you specify the **strip-group** command, the security appliance selects the tunnel group for user connections by obtaining the group name from the username presented by the VPN client. The security appliance then sends only the user part of the username for authorization/authentication. Otherwise (if disabled), the security appliance sends the entire username, including the realm.

Strip-realm processing removes the realm from the username when sending the username to the authentication or authorization server. If the command is enabled, the security appliance sends only the user part of the username authorization/authentication. Otherwise, the security appliance sends the entire username.

- 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, 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.
- 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 security appliance 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 security appliance monitor the continued presence of a remote peer and report its own presence to that peer. If the peer becomes unresponsive, the security appliance 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 security appliance and its remote peer must support a common form. This feature works with the following peers:

- 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 30-34](#).


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 tunnel groups or for particular tunnel groups.
- 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.

WebVPN Tunnel-Group Connection Parameters

The following attributes are specific to WebVPN connections:

- The authentication method, either AAA or certificate.
- The name of the 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 WebVPN.
- The DNS server-group name. The DNS server group specifies the DNS server name, domain name, name server, number of retries, and timeout values for a DNS server to use for a tunnel group.

- One or more group aliases; these are alternate names by which the server can refer to a tunnel group. At login, the user selects the group name from a dropdown menu.
- One or more group URLs. If you configure this parameter, users coming in on a specified URL need not select a group at login.
- A group policy that grants a WebVPN user access rights that are different from the default group policy.
- The name of the NetBIOS Name Service server (nbns-server) to use for CIFS name resolution.

Configuring Tunnel Groups

The following sections describe the contents and configuration of tunnel groups:

- [Default IPsec Remote Access Tunnel Group Configuration, page 30-5](#)
- [Specifying a Name and Type for the IPsec Remote Access Tunnel Group, page 30-6](#)
- [Configuring IPsec Remote-Access Tunnel Groups, page 30-6](#)
- [Configuring LAN-to-LAN Tunnel Groups, page 30-13](#)
- [Configuring WebVPN Tunnel Groups, page 30-17](#)
- [Customizing Login Windows for WebVPN Users, page 30-23](#)

You can modify the default tunnel groups, and you can configure a new tunnel group as any of the three tunnel-group types. If you don't explicitly configure an attribute in a tunnel group, that attribute gets its value from the default tunnel group. The default tunnel-group type is ipsec-ra. The subsequent parameters depend upon your choice of tunnel type. To see the current configured and default configuration of all your tunnel groups, including the default tunnel group, enter the **show running-config all tunnel-group** command.

Maximum Tunnel Groups

The maximum number of tunnel groups that a security appliance can support is a function of the maximum number of concurrent VPN sessions for the platform + 5. For example, an ASA5505 can support a maximum of 25 concurrent VPN sessions allowing for 30 tunnel groups (25+5). Attempting to add an additional tunnel group beyond the limit results in the following message: "ERROR: The limit of 30 configured tunnel groups has been reached."

[Table 30-1](#) specifies the maximum VPN sessions and connection profiles for each ASA platform.

Table 30-1 Maximum VPN Sessions and Connection Profiles Per ASA Platform

	5505 Base/ Security Plus	5510/Base/ Security Plus	5520	5540	5550
Maximum VPN Sessions	10/25	250	750	5000	5000
Maximum Connection Profiles	15/30	255	755	5005	5005

Default IPsec Remote Access Tunnel Group Configuration

The contents of the default remote-access tunnel group are as follows:

```

tunnel-group DefaultRAGroup type ipsec-ra
tunnel-group DefaultRAGroup general-attributes
no address-pool
authentication-server-group LOCAL
no authorization-server-group
no accounting-server-group
default-group-policy DfltGrpPolicy
no dhcp-server
no nac-authentication-server-group
no strip-realm
no password-management
no override-account-disable
no strip-group
no authorization-required
authorization-dn-attributes CN OU
tunnel-group DefaultRAGroup ipsec-attributes
no pre-shared-key
peer-id-validate req
no chain
no trust-point
isakmp keepalive threshold 100 retry 2
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 WebVPN tunnels share most of the same general attributes. IPsec LAN-to-LAN tunnels use a subset. Refer to the *Cisco Security Appliance Command Reference* for complete descriptions of all commands. The following sections describe, in order, how to configure IPsec remote-access tunnel groups, IPsec LAN-to-LAN tunnel groups, and WebVPN tunnel groups.

Configuring IPsec Remote-Access Tunnel Groups

Use an IPsec remote-access tunnel group when setting up a connection between a remote client and a central-site security appliance, using a hardware or software client. To configure an IPsec remote-access tunnel group, first configure the tunnel-group general attributes, then the IPsec remote-access attributes. An IPsec Remote Access VPN tunnel group applies only to remote-access IPsec client connections. To configure an IPsec remote-access tunnel group, see the following sections:

- [Specifying a Name and Type for the IPsec Remote Access Tunnel Group, page 30-6.](#)
- [Configuring IPsec Remote-Access Tunnel Group General Attributes, page 30-7.](#)
- [Configuring IPsec Remote-Access Tunnel Group IPsec Attributes, page 30-10.](#)

Specifying a Name and Type for the IPsec Remote Access Tunnel Group

Create the tunnel group, specifying its name and type, by entering the **tunnel-group** command. For an IPsec remote-access tunnel, the type is **ipsec-ra**

```
hostname(config)# tunnel-group tunnel_group_name type ipsec-ra
hostname(config)#
```

For example, to create an IPSec remote-access tunnel-group named TunnelGroup1, enter the following command:

```
hostname(config)# tunnel-group TunnelGroup1 type ipsec-ra
hostname(config)#
```

Configuring IPSec Remote-Access Tunnel Group General Attributes

To configure or change the tunnel group 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. 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)#
```

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 IPSec 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)#
```

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)#
```

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 following example sets DfltGrpPolicy as the name of the default group policy:

```
hostname(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.

```
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.

- 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:

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

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

```
hostname(config-group-policy)# no nac-authentication-server-group
hostname(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 security appliance uses the username and the group (if present) authentication. If you strip the group, the security appliance 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 qualifier 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.



Note The **strip-group** command, for tunnel group switching, does not work when MS-CHAPv2 is used for PPP authentication. This is due to a limitation on MS-CHAPv2 protocol. That is due to the hash computation during MS-CHAPv2 being bound to the username string.

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 security appliance 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 [“Setting the LDAP Server Type” section on page 13-7](#) 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 security appliance notifies the remote user at login that the user’s current password is about to expire or has expired. The security appliance 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 security appliance 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 security appliance 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 security appliance 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 30-24](#) for more information.

**Note**

The **radius-with-expiry** command, formerly configured as part of tunnel-group ipsec-ra configuration, is deprecated. The **password-management** command, entered in tunnel-group general-attributes mode, replaces it.

- 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:

```
hostname(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:

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

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

Configuring IPsec Remote-Access Tunnel Group IPsec Attributes

To configure the IPsec attributes for a remote-access tunnel group, do the following steps. The following description assumes that you have already created the IPsec remote-access tunnel group. IPsec remote-access tunnel groups have more attributes than IPsec LAN-to-LAN tunnel groups:

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

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

This command enters **tunnel-group ipsec-attributes** configuration mode, in which you configure the remote-access tunnel-group IPsec attributes.

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

```
hostname(config)# tunnel-group TG1 type ipsec-ra
hostname(config)# tunnel-group TG1 ipsec-attributes
hostname(config-tunnel-ipsec)#
```

- Step 2** Specify the preshared key to support IKE connections based on preshared keys. For example, the following command specifies the preshared key `xyzx` to support IKE connections for an IPSec remote access tunnel group:

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

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

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

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

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

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

- Step 4** Specify whether to

- Step 5** Specify whether to enable sending of a certificate chain. The following command includes the root certificate and any subordinate CA certificates in the transmission:

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

This attribute applies to all IPSec tunnel-group types.

- Step 6** Specify the name of a trustpoint that identifies the certificate to be sent to the IKE peer:

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

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

```
hostname(config-ipsec)# trust-point mytrustpoint
```

- Step 7** Specify the ISAKMP (IKE) keepalive threshold and the number of retries allowed.

```
hostname(config-tunnel-ipsec)# isakmp keepalive threshold <number> retry <number>
hostname(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 IKE keepalives, enter the **no** form of the **isakmp** command:

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

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

The default value for the **threshold** parameter 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 ("head end") should never initiate ISAKMP monitoring, enter the following command:

```
hostname(config-tunnel-ipsec)# isakmp keepalive threshold infinite
hostname(config-tunnel-ipsec)#
```

- Step 8** 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 security appliance 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 security appliance 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 tunnel group, 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 tunnel group called example-group:

```
hostname(config)# tunnel-group example-group type ipsec-ra
hostname(config)# tunnel-group example-group ipsec-attributes
hostname(config-tunnel-ipsec)# isakmp ikev1-user-authentication (inside) hybrid
hostname(config-tunnel-ipsec)#
```

Configuring IPSec Remote-Access Tunnel Group PPP Attributes

To configure the Point-to-Point Protocol attributes for a remote-access tunnel group, do the following steps. PPP attributes apply *only* to IPSec remote-access tunnel groups. The following description assumes that you have already created the IPSec remote-access tunnel group.

- 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:

```
hostname(config)# tunnel-group tunnel-group-name type ipsec-ra
hostname(config)# tunnel-group tunnel-group-name ppp-attributes
hostname(config-tunnel-ppp)#
```

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

```
hostname(config)# tunnel-group TG1 type ipsec-ra
hostname(config)# tunnel-group TG1 ppp-attributes
hostname(config-tunnel-ppp)#
```

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

- 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:

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

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

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

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

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

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

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

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

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

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

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

Configuring LAN-to-LAN Tunnel Groups

An IPsec LAN-to-LAN VPN tunnel group 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 tunnel groups, LAN-to-LAN tunnels have fewer parameters. To configure a LAN-to-LAN tunnel group, follow the steps in this section.

Default LAN-to-LAN Tunnel Group Configuration

The contents of the default LAN-to-LAN tunnel group 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 pre-shared-key
  peer-id-validate req
  no chain
  no trust-point
  isakmp keepalive threshold 10 retry 2
```

LAN-to-LAN tunnel groups have fewer parameters than remote-access tunnel groups, 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 tunnel group.

Specifying a Name and Type for a LAN-to-LAN Tunnel Group

To specify a name and a type for a tunnel group, 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 tunnel group named docs, enter the following command:

```
hostname(config)# tunnel-group docs type ipsec-l2l
hostname(config)#
```

Configuring LAN-to-LAN Tunnel Group General Attributes

To configure the tunnel group general attributes, do the following steps:

- Step 1** Enter tunnel-group general-attributes mode by specifying the general-attributes keyword:

```
hostname(config)# tunnel-group tunnel-group-name general-attributes
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 tunnel group named docs, enter the following command:

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

- Step 2** Specify the name of the accounting-server group, if any, to use:

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

For example, the following command specifies the use of the accounting-server group acctgserv1:

```
hostname(config-tunnel-general)# accounting-server-group acctgserv1
hostname(config-tunnel-general)#
```

- Step 3** Specify the name of the default group policy:

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

For example, the following command specifies that the name of the default group policy is MyPolicy:

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

Configuring LAN-to-LAN IPsec Attributes

To configure the IPsec attributes, do the following steps:

- Step 1** To configure the tunnel-group IPsec attributes, enter tunnel-group ipsec-attributes configuration mode by entering the tunnel-group command with the IPsec-attributes keyword.

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

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

```
hostname(config)# tunnel-group TG1 ipsec-attributes
hostname(config-tunnel-ipsec)#
```

The prompt changes to indicate that you are now in tunnel-group ipsec-attributes configuration mode.

- Step 2** Specify the preshared key to support IKE connections based on preshared keys.

```
hostname(config-tunnel-ipsec)# pre-shared-key key
hostname(config-tunnel-ipsec)#
```

For example, the following command specifies the preshared key XYZX to support IKE connections for an IPsec LAN-to-LAN tunnel group:

```
hostname(config-tunnel-ipsec)# pre-shared-key xyzx
hostname(config-tunnel-general)#
```

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

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

The available options are **req** (required), **cert** (if supported by certificate), and **nocheck** (do not check). The default is **req**. For example, the following command sets the peer-id-validate option to **nocheck**:

```
hostname(config-tunnel-ipsec)# peer-id-validate nocheck
hostname(config-tunnel-ipsec)#
```

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

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

You can apply this attribute to all tunnel-group types.

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

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

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

```
hostname(config-tunnel-ipsec)# trust-point mytrustpoint
hostname(config-tunnel-ipsec)#
```

You can apply this attribute to all tunnel-group types.

- Step 6** Specify the ISAKMP(IKE) keepalive threshold and the number of retries allowed. The **threshold** parameter specifies the number of seconds (10 through 3600) that the peer is allowed to idle before beginning keepalive monitoring. The **retry** parameter is the interval (2 through 10 seconds) between retries after a keepalive response has not been received. IKE keepalives are enabled by default. To disable IKE keepalives, enter the **no** form of the **isakmp** command:

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

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

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

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

To specify that the central site (“head end”) should never initiate ISAKMP monitoring, enter the following command:

```
hostname(config-tunnel-ipsec)# isakmp keepalive threshold infinite
hostname(config-tunnel-ipsec)#
```

- Step 7** Specify the ISAKMP hybrid authentication method, XAUTH or hybrid XAUTH.

You use **isakmp ikev1-user-authentication** command to implement hybrid XAUTH authentication when you need to use digital certificates for security appliance 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 security appliance 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 tunnel group, 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 tunnel group called example-group:

```
hostname(config)# tunnel-group example-group type ipsec-ra
hostname(config)# tunnel-group example-group ipsec-attributes
hostname(config-tunnel-ipsec)# isakmp ikev1-user-authentication (inside) hybrid
hostname(config-tunnel-ipsec)#
```

Configuring WebVPN Tunnel Groups

A WebVPN tunnel group applies only to WebVPN connections. The tunnel-group general attributes for WebVPN tunnel groups are the same as those of IPsec remote-access tunnel groups, except that the tunnel-group type is `webvpn` and the `strip-group` and `strip-realm` commands do not apply. You define the WebVPN-specific attributes separately. The following sections describe how to configure WebVPN tunnel groups.

Specifying a Name and Type for a WebVPN Tunnel Group

Create the tunnel group, specifying its name and type by entering the `tunnel-group` command in global configuration mode. For an IPsec remote-access tunnel, the type is `webvpn`.

```
hostname(config)# tunnel-group tunnel_group_name type webvpn
hostname(config)#
```

For example, to create a WebVPN tunnel-group named `TunnelGroup3`, enter the following command:

```
hostname(config)# tunnel-group TunnelGroup3 type webvpn
hostname(config)#
```

Configuring WebVPN Tunnel-Group General Attributes

To configure or change the tunnel group 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. 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

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.

```
hostname(config-tunnel-general)# authorization-required
hostname(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:

```
hostname(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:

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

```
hostname(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 31, “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 security appliance 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 [“Setting the LDAP Server Type”](#) section on page 13-7 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 security appliance notifies the remote user at login that the user's current password is about to expire or has expired. The security appliance 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 security appliance 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 security appliance 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 30-24](#) for more information.

- Step 10** Specifying this command with the number of days set to 0 disables this command. The security appliance 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 WebVPN Tunnel-Group WebVPN Attributes

To configure the parameters specific to a WebVPN tunnel group, follow the steps in this section.

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

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

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

```
hostname(config)# tunnel-group sales webvpn-attributes
hostname(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.

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

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

```
hostname(config-tunnel-webvpn)# authentication aaa certificate
hostname(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 WebVPN.

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:

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

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

```
hostname(config-username-webvpn)# customization value blueborder
hostname(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 WebVPN customization named “123” that defines a password prompt. The example then defines a WebVPN tunnel-group named “test” and uses the **customization** command to specify the use of the WebVPN customization named “123”:

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

Step 3 The security appliance queries NetBIOS name servers to map NetBIOS names to IP addresses. WebVPN requires NetBIOS to access or share files on remote systems. WebVPN 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 security appliance 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 security appliance, or reachable from that network. You must specify the timeout interval before the number of retries:

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

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

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

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

```
hostname(config-tunnel-webvpn)# group-alias QA enable
hostname(config-tunnel-webvpn)# group-alias Devtest enable
hostname(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 security appliance 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 tunnel group, then the security appliance automatically selects the associated tunnel group 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 tunnel group.

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 security appliance verifies the uniqueness of the URL or address before accepting the URL or address for a tunnel group.

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

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

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

```
hostname(config)# tunnel-group RadiusServer type webvpn
hostname(config)# tunnel-group RadiusServer general-attributes
hostname(config-tunnel-general)# authentication server-group RADIUS
hostname(config-tunnel-general)# accounting-server-group RADIUS
hostname(config-tunnel-general)# tunnel-group RadiusServer webvpn-attributes
```

```
hostname(config-tunnel-webvpn)# group-alias "Cisco Remote Access" enable
hostname(config-tunnel-webvpn)# group-url http://www.cisco.com enable
hostname(config-tunnel-webvpn)# group-url http://192.168.10.10 enable
hostname(config-tunnel-webvpn)#
```

For a more extensive example, see [Customizing Login Windows for WebVPN Users, page 30-23](#).

- Step 6** To specify the DNS server to use for a WebVPN tunnel group, enter the **dns-group** command. The default value is DefaultDNS:

```
hostname(config-tunnel-webvpn)# dns-group {hostname | ip_address}
hostname(config-tunnel-webvpn)#
```

The **dns-group** command resolves the hostname to the appropriate DNS server for the tunnel group. For example, to specify the use of the DNS server named `server1`, enter the following command:

```
hostname(config)# name 10.10.10.1 server1
hostname(config-tunnel-webvpn)# dns-group server1
hostname(config-tunnel-webvpn)#
```

- Step 7** (Optional) To specify 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,” use the **hic-fail-group-policy** command. The default value is DfltGrpPolicy.

```
hostname(config-tunnel-webvpn)# hic-fail-group-policy name
hostname(config-tunnel-webvpn)#
```

Name is the name of a group policy created for a WebVPN tunnel group.

This policy is an alternative group policy to differentiate access rights for the following CSD clients:

- Clients that match a CSD location entry set to “Use Failure Group-Policy.”
- Clients that match a CSD location entry set to “Use Success Group-Policy, if criteria match,” and then fail to match the configured Group-Based Policy criteria. For more information, see the *Cisco Secure Desktop Configuration Guide for Cisco ASA 5500 Series Administrators*.

The following example specifies an alternative group policy named `group2`:

```
hostname(config-tunnel-webvpn)# hic-fail-group-policy group2
hostname(config-tunnel-webvpn)#
```



Note The security appliance does not use this attribute if you set the VPN feature policy to “Always use Success Group-Policy.”

For more information, see the *Cisco Secure Desktop Configuration Guide for Cisco ASA 5500 Series Administration Guide*.

Customizing Login Windows for WebVPN Users

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

- Step 1** In webvpn mode, define a WebVPN customization, 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 security appliance and saved the configuration. See the WebVPN chapter for details.

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

- Step 2** In global configuration mode, set up a username and associate with it the WebVPN customization you've just defined:

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

- Step 3** In global configuration mode, create a WebVPN tunnel-group named sales:

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

- Step 4** Specify that you want to use the salesgui customization for this tunnel group:

```
hostname# tunnel-group sales webvpn-attributes
hostname(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 security appliance; for example, if the security appliance has the IP address 192.168.3.3, set the group URL to https://192.168.3.3:

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

If a port number is required for a successful login, include the port number, preceded by a colon. The security appliance maps this URL to the sales tunnel group 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 security appliance 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 “[Setting the LDAP Server Type](#)” section on page 13-7 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 security appliance. 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 security appliance and configured the corresponding password management attributes. The specific steps in the following sections refer to Active Directory terminology under Windows 2000.

- [Using Active Directory to Force the User to Change Password at Next Logon, page 30-25.](#)
- [Using Active Directory to Specify Maximum Password Age, page 30-27.](#)
- [Using Active Directory to Override an Account Disabled AAA Indicator, page 30-28](#)
- [Using Active Directory to Enforce Password Complexity, page 30-30.](#)

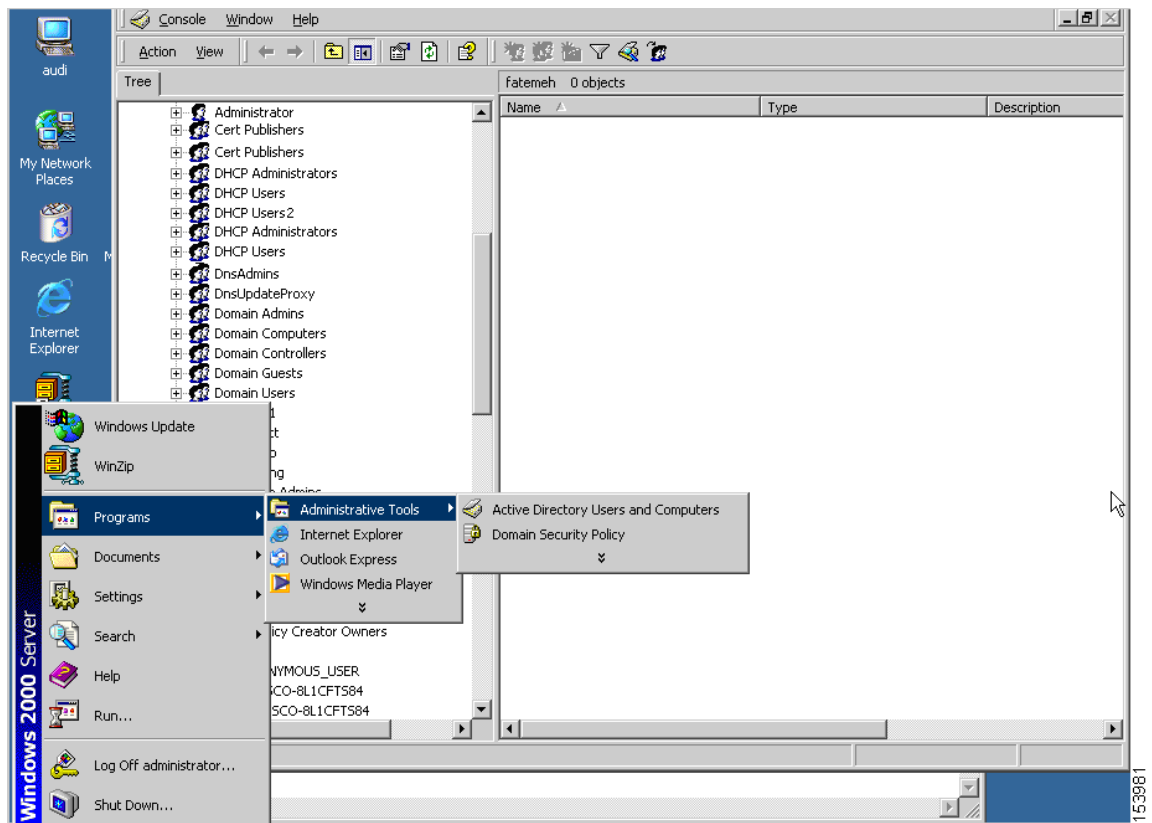
The following sections assume that you are using an LDAP directory server for authentication.

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

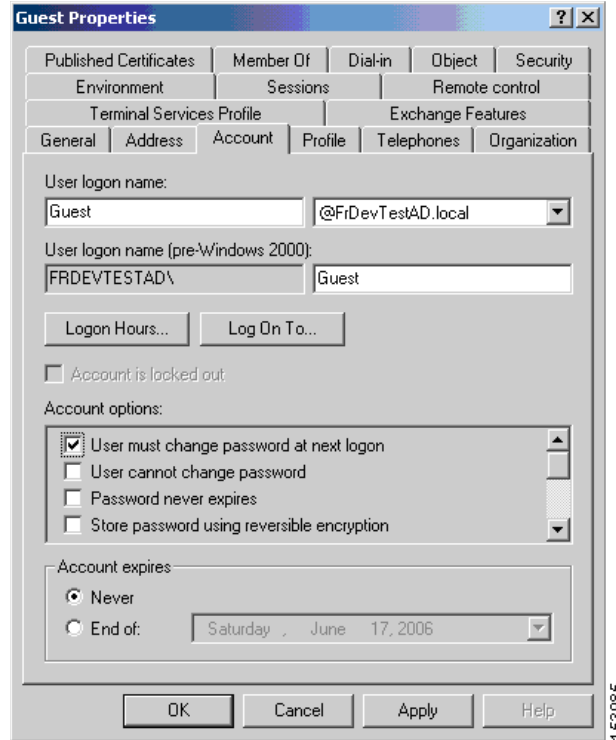
To force a user to change the user password at the next logon, specify the **password-management** command in tunnel-group general-attributes configuration mode on the security appliance and do the following steps under Active Directory:

-
- Step 1** Select Start > Programs > Administrative Tools > Active Directory Users and Computers ([Figure 30-1](#)).

Figure 30-1 Active Directory—Administrative Tools Menu



- Step 2** Right-click Username > Properties > Account.
- Step 3** Check the check box for User must change password at next logon ([Figure 30-2](#)).

Figure 30-2 Active Directory—User Must Change Password at Next Logon

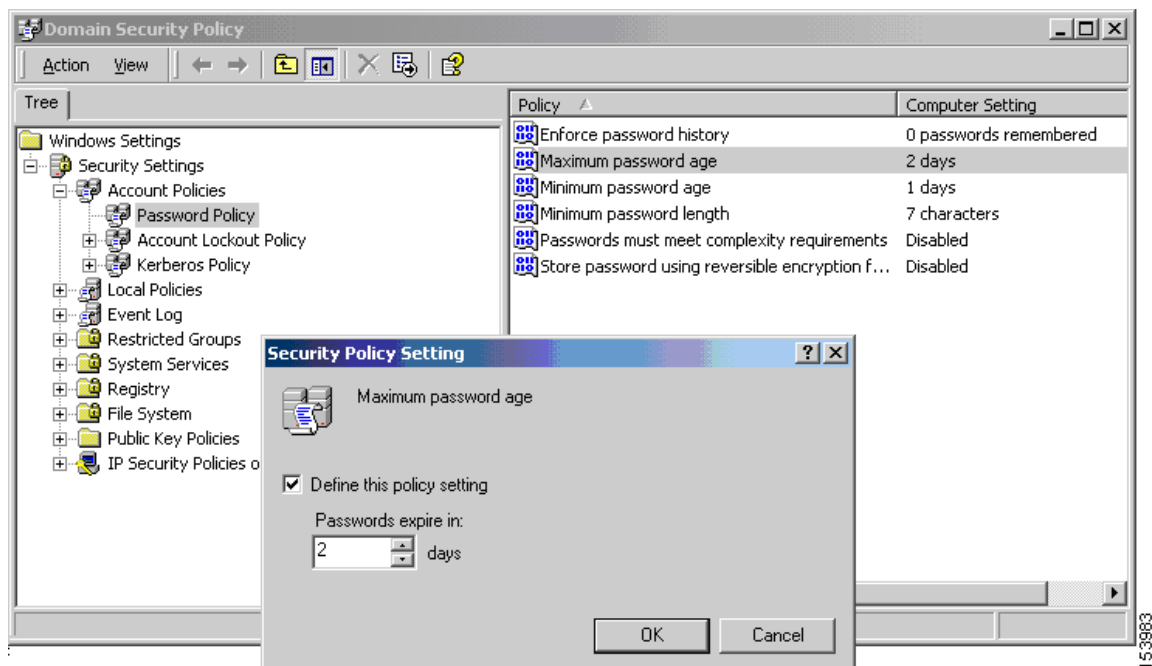
The next time this user logs on, the security appliance displays the following prompt: “New password required. Password change required. You must enter a new password with a minimum length n to continue.” You can set the minimum required password length, n , as part of the Active Directory configuration 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 security appliance and do the following steps under Active Directory:

- Step 1** Select Start > Programs > Administrative Tools > Domain Security Policy > Windows Settings > Security Settings > Account Policies > Password Policy.
- Step 2** Double-click Maximum password age. This opens the Security Policy Setting dialog box.
- Step 3** Check the Define this policy setting check box and specify the maximum password age, in days, that you want to allow.

Figure 30-3 Active Directory—Maximum Password Age



Note The **radius-with-expiry** command, formerly configured as part of tunnel-group ipsec-ra 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

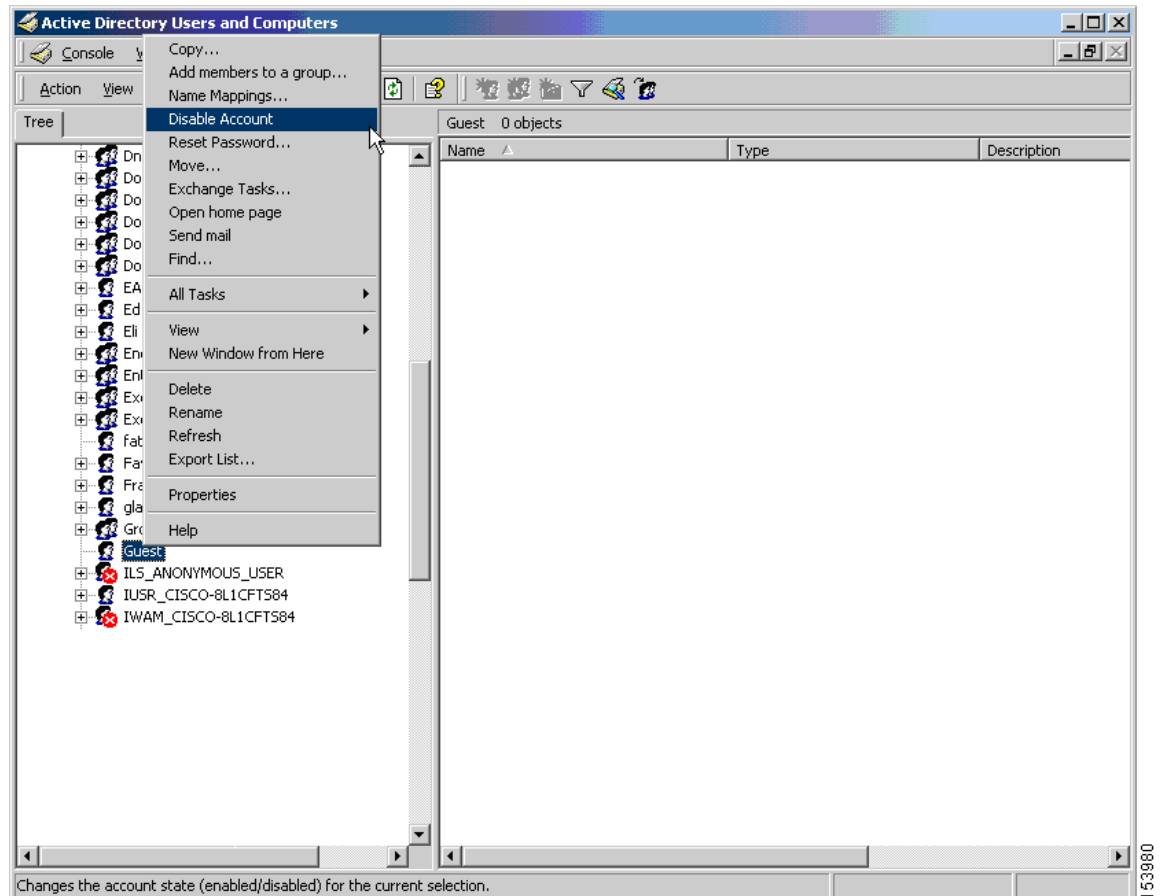
To override an account-disabled indication from a AAA server, specify the **override-account-disable** command in tunnel-group general-attributes configuration mode on the security appliance and do the following steps under Active Directory:



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 30-4 Active Directory—Override Account Disabled



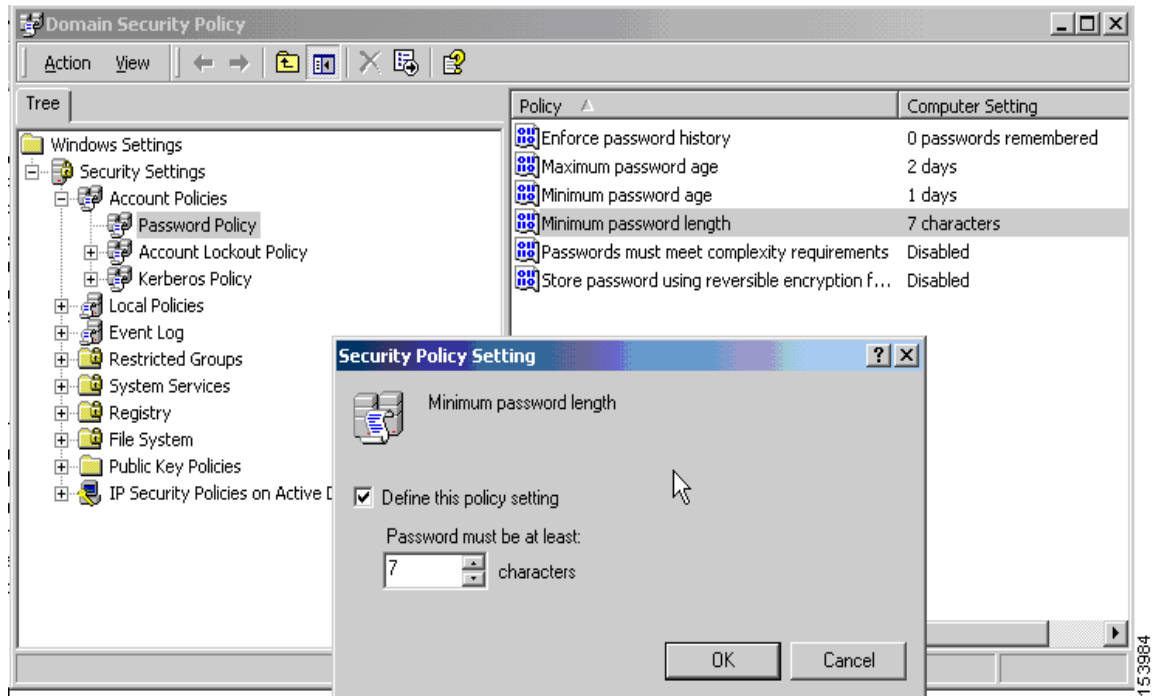
The user should be able to log on successfully, even though a AAA server provides an account-disabled indicator.

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 security appliance and do 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. This opens the Security Policy Setting dialog box.
- Step 4** Check the Define this policy setting check box and specify the minimum number of characters that the password must contain.

Figure 30-5 Active Directory—Minimum Password Length

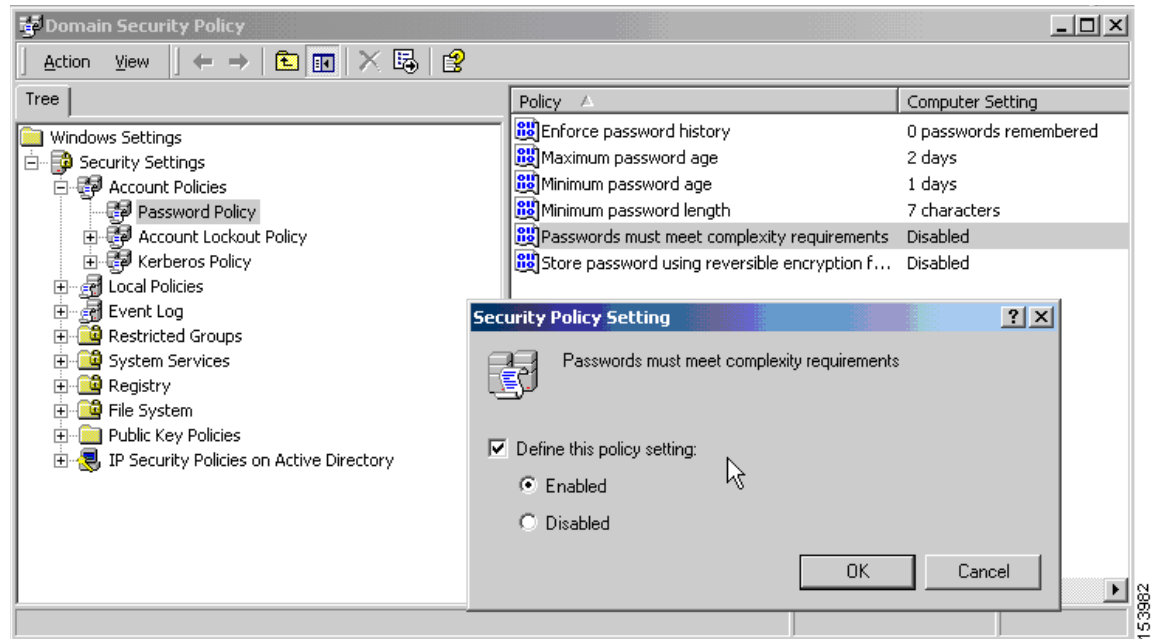


Using Active Directory to Enforce Password Complexity

To enforce complex passwords—for example, to require that a password contain upper- and lowercase letters, numbers, and special characters—specify the **password-management** command in tunnel-group general-attributes configuration mode on the security appliance and do 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 30-6 Active Directory—Enforce Password Complexity



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.

Group Policies

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

- [Default Group Policy, page 30-32](#)
- [Configuring Group Policies, page 30-34](#)

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 tunnel group 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 security appliance 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 security appliance'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
- WebVPN functions
- Connection settings

Default Group Policy

The security appliance 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 security appliance, but this default group policy does not take effect unless you configure the security appliance to use it. When you configure other group policies, any attribute that you do not explicitly specify takes its value from the default group policy. To view the default group policy, enter the following command:

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

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

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



Note

The default group policy is always internal. Despite the fact that the command syntax is `hostname(config)# group-policy DfltGrpPolicy {internal | external}`, you cannot change the type to external.

To change any of the attributes of the default group policy, use the **group-policy attributes** command to enter attributes mode, then specify the commands to change whatever attributes that you want to modify:

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



Note

The attributes mode applies only to internal group policies.

The default group policy, `DfltGrpPolicy`, that the security appliance provides is as follows:

```
group-policy DfltGrpPolicy internal
group-policy DfltGrpPolicy attributes
  banner none
  wins-server none
  dns-server none
  dhcp-network-scope none
  vpn-access-hours none
  vpn-simultaneous-logins 2000
  vpn-idle-timeout none
  vpn-session-timeout none
  vpn-filter none
```



```

vpn-tunnel-protocol IPSec webvpn
password-storage enable
ip-comp disable
re-xauth disable
group-lock none
pfs disable
ipsec-udp disable
ipsec-udp-port 10000
split-tunnel-policy tunnelall
split-tunnel-network-list none
default-domain none
split-dns none
intercept-dhcp 255.255.255.255 disable
secure-unit-authentication disable
user-authentication disable
user-authentication-idle-timeout 30
ip-phone-bypass disable
leap-bypass disable
nem disable
backup-servers keep-client-config
msie-proxy server none
msie-proxy method no-modify
msie-proxy except-list none
msie-proxy local-bypass disable
nac disable
nac-sq-period 300
nac-reval-period 36000
nac-default-acl none
address-pools value vpn_users
client-firewall none
client-access-rule none
webvpn
  functions url-entry file-access file-browsing
  html-content-filter none
  homepage none
  keep-alive-ignore 4
  http-comp gzip
  filter none
  url-list value MyURLs
  customization value DfltCustomization
  port-forward none
  port-forward-name value Application Access
  sso-server none
  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
  svc none
  svc keep-installer none
  svc keepalive none
  svc rekey time none
  svc rekey method none
  svc dpd-interval client none
  svc dpd-interval gateway none
  svc compression deflate
no vpn-nac-exempt
hostname(config-group-policy)#

```

You can modify the default group policy, and you can also create one or more group policies specific to your environment.

Configuring Group Policies

A group policy can apply to any kind of tunnel. In each case, if you do not explicitly define a parameter, the group takes the value from the default group policy. To configure a group policy, follow the steps in the subsequent sections.

Configuring an External Group Policy

External group policies take their attribute values from the external server that you specify. For an external group policy, you must identify the AAA server group that the security appliance 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 security appliance refer to user names on the RADIUS server. In other words, if you configure external group X on the security appliance, 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 security appliance. 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 security appliance supports user authorization on an external LDAP or RADIUS server. Before you configure the security appliance to use an external server, you must configure the server with the correct security appliance authorization attributes and, from a subset of these attributes, assign specific permissions to individual users. Follow the instructions in [Appendix E, “Configuring an External Server for Authorization and Authentication”](#) to configure your external server.

To configure an external group policy, do the following steps specify a name and type for the group policy, along with the server-group name and a password:

```
hostname(config)# group-policy group_policy_name type server-group server_group_name
password server_password
hostname(config)#
```

**Note**

For an external group policy, RADIUS is the only supported AAA server type.

For example, the following command creates an external group policy named ExtGroup that gets its attributes from an external RADIUS server named ExtRAD and specifies that the password to use when retrieving the attributes is newpassword:

```
hostname(config)# group-policy ExtGroup external server-group ExtRAD password newpassword
hostname(config)#
```

**Note**

You can configure several vendor-specific attributes (VSAs), as described in [Appendix E, “Configuring an External Server for Authorization and Authentication”](#). If a RADIUS server is configured to return the Class attribute (#25), the security appliance 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 security appliance—for example, OU=Finance.

Configuring an Internal Group Policy

To configure an internal group policy, specify a name and type for the group policy:

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

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

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

The default type is **internal**.

You can initialize the attributes of an internal group policy to the values of a preexisting group policy by appending the keyword **from** and specifying the name of the existing policy:

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

Configuring Group Policy Attributes

For internal group policies, you can specify particular attribute values. To begin, enter group-policy attributes mode, by entering the **group-policy attributes** command in global configuration mode.

```
hostname(config)# group-policy name attributes
hostname(config-group-policy)#
```

The prompt changes to indicate the mode change. The group-policy-attributes mode lets you configure attribute-value pairs for a specified group policy. In group-policy-attributes mode, explicitly configure the attribute-value pairs that you do not want to inherit from the default group. The commands to do this are described in the following sections.

Configuring WINS and DNS Servers

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

Step 1 Specify the primary and secondary WINS servers:

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

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

Step 2 Specify the primary and secondary DNS servers:

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

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, and 10.10.10.30 for the group policy named FirstGroup:

```
hostname(config)# group-policy FirstGroup attributes
hostname(config-group-policy)# dns-server value 10.10.10.15 10.10.10.30
hostname(config-group-policy)#
```

Step 3 Configure the DHCP network scope:

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

DHCP scope specifies the range of IP addresses (that is, a subnetwork) that the security appliance 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:

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

Configuring VPN-Specific Attributes

Follow the steps in this section to configure attributes that set the values of VPN attributes. These attributes control the access hours, the number of simultaneous logins allowed, the timeouts, the name of the ACL to use for VPN connections, and the tunnel protocol:

Step 1 Set the VPN access hours. To do this, you associate a group policy with a configured time-range policy, using the **vpn-access-hours** command in group-policy configuration mode.

```
hostname(config-group-policy)# vpn-access-hours value {time-range | none}
```

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.

The time-range variable is the name of a set of access hours defined in global configuration mode using the **time-range** command. The following example shows how to associate the group policy named FirstGroup with a time-range policy called 824:

```
hostname(config)# group-policy FirstGroup attributes
hostname(config-group-policy)# vpn-access-hours value 824
```

- Step 2** Specify the number of simultaneous logins allowed for any user, using the **vpn-simultaneous-logins** command in group-policy configuration mode.

```
hostname(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:

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



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

- Step 3** Configure the user timeout period by entering the **vpn-idle-timeout** command in group-policy configuration mode or in username configuration mode:

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

The minimum time is 1 minute, and the maximum time is 35791394 minutes. The default is 30 minutes. If there is no communication activity on the connection in this period, the security appliance terminates the connection.

A group policy can inherit this value from another group policy. To prevent inheriting a value, enter the **none** keyword instead of specifying a number of minutes with this command. The none keyword also permits an unlimited idle timeout period. It sets the idle timeout to a null value, thereby disallowing an idle timeout.

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

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

- Step 4** 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.

```
hostname(config-group-policy)# vpn-session-timeout {minutes | none}
hostname(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 security appliance 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:

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

- Step 5** Specify the name of the ACL to use for VPN connections, using the **vpn-filter** command in group policy mode. (You can also configure this attribute in username mode, in which case the value configured under username supersedes the group-policy value.)

```
hostname(config-group-policy)# vpn-filter {value ACL name | none}
hostname(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 access list and sets a null value, thereby disallowing an access list.

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

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

- Step 6** Specify the VPN tunnel type (IPSec or WebVPN) for this group policy.

```
hostname(config-group-policy)# vpn-tunnel-protocol {webvpn | IPSec | l2tp-ipsec}
hostname(config-group-policy)#
```

The default is IPSec. To remove the attribute from the running configuration, enter the **no** form of this command.

```
hostname(config-group-policy)# no vpn-tunnel-protocol [webvpn | IPSec | l2tp-ipsec]
hostname(config-group-policy)#
```

The parameter values for this command follow:

- **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 VPN services to remote users via an HTTPS-enabled web browser, and does not require a client.
- **l2tp-ipsec**—Negotiates an IPSec tunnel for an L2TP connection

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 tunneling mode for the group policy named FirstGroup:

```
hostname(config)# group-policy FirstGroup attributes
hostname(config-group-policy)# vpn-tunnel-protocol IPSec
hostname(config-group-policy)#
```

Configuring Security Attributes

The attributes in this section specify certain security settings for the group:

- 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.

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

```
hostname(config-group-policy)# no password-storage
hostname(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:

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

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

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

```
hostname(config-group-policy)# no ip-comp
hostname(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 security appliance. 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 rekey by using the **re-xauth** command with the **enable** keyword in group-policy configuration mode. If you enable reauthentication on IKE rekey, the security appliance prompts the user to enter a username and password during initial Phase 1 IKE negotiation and also prompts for user authentication whenever an IKE rekey occurs. Reauthentication provides additional security.

If the configured rekey interval is very short, users might find the repeated authorization requests inconvenient. To avoid repeated authorization requests, disable reauthentication. To check the configured rekey 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 rekey, enter the **disable** keyword. Reauthentication on IKE rekey is disabled by default.

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

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

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



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

Step 4 Specify whether to restrict remote users to access only through the tunnel group, using the **group-lock** command in **group-policy** configuration mode.

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

The *tunnel-grp-name* variable specifies the name of an existing tunnel group that the security appliance 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 tunnel group to which the user is assigned. If it is not, the security appliance prevents the user from connecting. If you do not configure group-lock, the security appliance 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.

Step 5 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.

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

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

Configuring the 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/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:

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

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

```
hostname(config)# group-policy FirstGroup attributes
hostname(config-group-policy)# banner value Welcome to Cisco Systems 7.0.
```

Configuring IPSec-UDP Attributes

IPSec over UDP, sometimes called IPSec through NAT, lets a Cisco VPN client or hardware client connect via UDP to a security appliance 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 security appliance 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:

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

To use IPSec over UDP, you must also configure the **ipsec-udp-port** command, as described below.

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:

```
hostname(config)# group-policy FirstGroup attributes
hostname(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 security appliance 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.

```
hostname(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:

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

Configuring Split-Tunneling Attributes

Split tunneling lets a remote-access IPSec client conditionally direct packets over an IPSec tunnel in encrypted form or to a network interface in clear text form. With split tunneling enabled, packets not bound for destinations on the other side of the IPSec tunnel do not have to be encrypted, sent across the tunnel, decrypted, and then routed to a final destination. This command applies this split tunneling policy to a specific network.

Setting the Split-Tunneling Policy

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

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

The default is to tunnel all traffic. To set a split tunneling policy, enter the **split-tunnel-policy** command in group-policy configuration mode. To remove the **split-tunnel-policy** attribute from the running configuration, enter the **no** form of this command. This enables inheritance of a value for split tunneling from another group policy.

The **excludespecified** keyword defines a list of networks to which traffic goes in the clear. This feature is useful for remote users who want to access devices on their local network, such as printers, while they are connected to the corporate network through a tunnel. This option applies only to the Cisco VPN client.

The **tunnelall** keyword specifies that no traffic goes in the clear or to any other destination than the security appliance. This, in effect, disables split tunneling. Remote users reach Internet networks through the corporate network and do not have access to local networks. This is the default option.

The **tunnelspecified** keyword tunnels all traffic from or to the specified networks. This option enables split tunneling. It lets you create a network list of addresses to tunnel. Data to all other addresses travels in the clear and is routed by the remote user's Internet service provider.



Note

Split tunneling is primarily 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:

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

Creating a Network List for Split-Tunneling

Create a network list for split tunneling using the **split-tunnel-network-list** command in group-policy configuration mode.

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

Split tunneling network lists distinguish networks that require traffic to travel across the tunnel from those that do not require tunneling. The security appliance makes split tunneling decisions on the basis of a network list, which is an ACL that consists of a list of addresses on the private network. Only standard-type ACLs are allowed.

The **value** *access-list name* parameter identifies an access list that enumerates the networks to tunnel or not tunnel.

The **none** keyword indicates that there is no network list for split tunneling; the security appliance 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.

The following example shows how to set a network list called FirstList for the group policy named FirstGroup:

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

Configuring Domain Attributes for Tunneling

You can specify a default domain name for tunneled packets or a list of domains to be resolved through the split tunnel. The following sections describe how to set these domains.

Defining a Default Domain Name for Tunneled Packets

The security appliance passes the default domain name to the IPsec client to append to DNS queries that omit the domain field. When there are no default domain names, users inherit the default domain name in the default group policy. To specify the default domain name for users of the group policy, enter the **default-domain** command in group-policy configuration mode. To delete a domain name, enter the **no** form of this command.

```
hostname(config-group-policy)# default-domain {value domain-name | none}
hostname(config-group-policy)# no default-domain [domain-name]
```

The **value** *domain-name* parameter identifies the default domain name for the group. To specify that there is no default domain name, enter the **none** keyword. This command sets a default domain name with a null value, which disallows a default domain name and prevents inheriting a default domain name from a default or specified group policy.

To delete all default domain names, enter the **no default-domain** command without arguments. This command deletes all configured default domain names, including a null list if you created one by entering the **default-domain** command with the **none** keyword. The **no** form allows inheriting a domain name.

The following example shows how to set a default domain name of FirstDomain for the group policy named FirstGroup:

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

Defining a List of Domains for Split Tunneling

Enter a list of domains to be resolved through the split tunnel. Enter the **split-dns** command in group-policy configuration mode. To delete a list, enter the **no** form of this command.



Note

The AnyConnect VPN client and the SSL VPN Client do not support split DNS.

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 security appliance resolves through the split tunnel. The **none** keyword indicates that there is no split DNS list. It also sets a split DNS list with a null value, thereby disallowing a split DNS list, and prevents inheriting a split DNS list from a default or specified group policy. The syntax of the command is as follows:

```
hostname(config-group-policy)# split-dns {value domain-name1 [domain-name2...
domain-nameN] | none}
hostname(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:

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

Configuring DHCP Intercept

A Microsoft XP anomaly results in the corruption of domain names if split tunnel options exceed 255 bytes. To avoid this problem, the security appliance 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 security appliance. The security appliance replies directly to the Microsoft Windows XP client DHCP Inform message, providing that client with the subnet mask, domain name, and classless static routes for the tunnel IP address. For Windows clients prior to Windows XP, DHCP Intercept provides the domain name and subnet mask. This is useful in environments in which using a DHCP server is not advantageous.

The **intercept-dhcp** command enables or disables DHCP intercept. The syntax of this command is as follows:

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

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

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

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

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

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

Configuring Attributes for VPN Hardware Clients

The commands in this section enable or disable secure unit authentication and user authentication, and 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 tunnel group the hardware client(s) use. If you require secure unit authentication on the primary security appliance, 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.

```
hostname(config-group-policy)# secure-unit-authentication {enable | disable}  
hostname(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:

```
hostname(config)# group-policy FirstGroup attributes  
hostname(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.

```
hostname(config-group-policy)# user-authentication {enable | disable}
hostname(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 security appliance, 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:

```
hostname(config)# group-policy FirstGroup attributes
hostname(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 security appliance terminates the client's access:

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



Note

The **user-authentication-idle-timeout** command terminates only the client's access through the VPN tunnel, not the VPN tunnel itself.

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:

```
hostname(config)# group-policy FirstGroup attributes
hostname(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:

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

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:

```
hostname(config-group-policy)# leap-bypass {enable | disable}
hostname(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:

```
hostname(config)# group-policy FirstGroup attributes
hostname(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 security appliance. PAT does not apply. Therefore, devices behind the security appliance 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:

```
hostname(config-group-policy)# nem {enable | disable}
hostname(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:

```
hostname(config)# group-policy FirstGroup attributes
hostname(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 security appliance is unavailable. When you configure backup servers, the security appliance 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 security appliance.

Configure backup servers either on the client or on the primary security appliance. If you configure backup servers on the security appliance, it pushes the backup server policy to the clients in the group, replacing the backup server list on the client if one is configured.



Note

If you are using hostnames, it is wise to have backup DNS and WINS servers on a separate network from that of the primary DNS and WINS servers. Otherwise, if clients behind a hardware client obtain DNS and WINS information from the hardware client via DHCP, and the connection to the primary server is lost, and the backup servers have different DNS and WINS information, clients cannot be updated until the DHCP lease expires. In addition, if you use hostnames and the DNS server is unavailable, significant delays can occur.

To configure backup servers, enter the **backup-servers** command in group-policy configuration mode:

```
hostname(config-group-policy)# backup-servers {server1 server2... server10 |
clear-client-config | keep-client-config}
```

To remove a backup server, enter the **no** form of this command with the backup server specified. To remove the backup-servers attribute from the running configuration and enable inheritance of a value for backup-servers from another group policy, enter the **no** form of this command without arguments.

```
hostname(config-group-policy)# no backup-servers [server1 server2... server10 |
clear-client-config | keep-client-config]
```

The **clear-client-config** keyword specifies that the client uses no backup servers. The security appliance pushes a null server list.

The **keep-client-config** keyword specifies that the security appliance 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 security appliance is unavailable. This list identifies servers by IP address or hostname. The list can be 500 characters long, and it can contain up to 10 entries.

The following example shows how to configure backup servers with IP addresses 10.10.10.1 and 192.168.10.14, for the group policy named FirstGroup:

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


Configuring Microsoft Internet Explorer Client Parameters

The following commands configure the proxy server parameters for a Microsoft Internet Explorer client.

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

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

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

```
hostname(config-group-policy)# no msie-proxy server
hostname(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 Microsoft Internet Explorer proxy server, using port 880, for the group policy named FirstGroup:

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

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

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

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

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

The available methods are as follows:

- **auto-detect**—Enables the use of automatic proxy server detection in Internet Explorer for the client PC.
- **no-modify**—Leaves the HTTP browser proxy server setting in Internet Explorer unchanged for this client PC.
- **no-proxy**—Disables the HTTP proxy setting in Internet Explorer for the client PC.
- **use-server**—Sets the HTTP proxy server setting in Internet Explorer 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 Microsoft Internet Explorer proxy setting for the group policy named FirstGroup:

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

The following example configures the Microsoft Internet Explorer proxy setting for the group policy named FirstGroup to use the server QAserver, port 1001 as the server for the client PC:

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

- Step 3** Configure Microsoft Internet Explorer browser proxy exception list settings for a local bypass on the client PC 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 in Internet Explorer.

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

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

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

- **value** *server:port*—Specifies the IP address or name of an MSIE server and port that is applied for this client PC. 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 Microsoft Internet Explorer proxy exception list, consisting of the server at IP address 192.168.20.1, using port 880, for the group policy named FirstGroup:

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

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

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

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

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

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

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

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

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 security appliance 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.

The following parameters let you configure Network Admission Control settings for the default group policy or an alternative group policy.

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:

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

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

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

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

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

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

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

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

The following example changes the revalidation timer to 86400 seconds:

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

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

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

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

```
hostname(config-group-policy) # no nac-default-acl [acl-name | none]
hostname(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 security appliance 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 security appliance 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:

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

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

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

```
hostname(config-group-policy) # nac-default-acl none
hostname(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** 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.

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

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

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

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

The syntax elements for these commands are as follows:

- *acl-name*—Name of the ACL present in the security appliance 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:

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

The following example exempts all hosts running Windows 98 that match an ACE in the ACL named **acl-1**:

```
hostname(config-group-policy)# vpn-nac-exempt os "Windows 98" filter acl-1
hostname(config-group-policy)
```

The following example adds the same entry to the exemption list, but disables it:

```
hostname(config-group-policy)# vpn-nac-exempt os "Windows 98" filter acl-1 disable
hostname(config-group-policy)
```

The following example removes the same entry from the exemption list, regardless of whether it is disabled:

```
hostname(config-group-policy)# no vpn-nac-exempt os "Windows 98" filter acl-1
hostname(config-group-policy)
```

The following example disables inheritance and specifies that all hosts will be subject to posture validation:

```
hostname(config-group-policy) # no vpn-nac-exempt none
hostname(config-group-policy)
```

The following example removes all entries from the exemption list:

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

Step 5 Enable or disable Network Admission Control by entering the following command:

```
hostname(config-group-policy) # nac {enable | disable}
hostname(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:

```
hostname(config-group-policy) # no nac [enable | disable]
hostname(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 security appliance 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:

```
hostname(config-group-policy) # nac enable
hostname(config-group-policy) #
```

Configuring Address Pools

Configure a list of address pools for allocating addresses to remote clients by entering the **address-pools** command in group-policy attributes configuration mode:

```
hostname(config-group-policy) # address-pools value address_pool1 [...address_pool6]
hostname(config-group-policy) #
```

The address-pools settings in this command override the local pool settings in the group. 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 security appliance allocates addresses from these pools in the order in which the pools appear in this command.

To remove the attribute from the group policy and enable inheritance from other sources of group policy, use the **no** form of this command:

```
hostname(config-group-policy) # no address-pools value address_pool1 [...address_pool6]
hostname(config-group-policy) #
```

The command **address-pools none** disables this attribute from being inherited from other sources of policy, such as the DefaultGrpPolicy:

```
hostname(config-group-policy) # address-pools none
hostname(config-group-policy) #
```

The command **no address pools none** removes the **address-pools none** command from the configuration, restoring the default value, which is to allow inheritance.

```
hostname(config-group-policy)# no address-pools none
hostname(config-group-policy)#
```

The syntax elements of this command are as follows:

- **address_pool**—Specifies the name of the address pool configured with the **ip local pool** command. You can specify up to 6 local address pools.
- **none**—Specifies that no address pools are configured and disables inheritance from other sources of group policy.
- **value**—Specifies a list of up to 6 address pools from which to assign addresses.

The following example entered in config-general configuration mode, configures pool 1 and pool20 as lists of address pools to use for allocating addresses to remote clients for GroupPolicy1:

```
hostname(config)# ip local pool pool 192.168.10.1-192.168.10.100 mask 255.255.0.0
hostname(config)# ip local pool pool20 192.168.20.1-192.168.20.200 mask 255.255.0.0
hostname(config)# group-policy GroupPolicy1 attributes
hostname(config-group-policy)# address-pools value pool1 pool20
hostname(config-group-policy)#
```

Configuring Firewall Policies

A *firewall* isolates and protects a computer from the Internet by inspecting each inbound and outbound individual packet of data to determine whether to allow or 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 PC, and thereby the corporate network, from intrusions by way of the Internet or the user's local LAN. Remote users connecting to the security appliance with the VPN client can choose the appropriate firewall option.

Set personal firewall policies that the security appliance 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 window, 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 security appliance. (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 security appliance.) 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 security appliance, 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 security appliance pushes this policy down to the VPN client. The VPN client then in turn passes the policy to the local firewall, which enforces it.

Supporting a Zone Labs Integrity Server

This section introduces the Zone Labs Integrity Server, also called Check Point Integrity Server, and presents an example procedure for configuring the security appliance 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 will not be granted access to the private network protected by the Integrity Server and security appliance.

This section includes the following topics:

- [Overview of Integrity Server and Security Appliance Interaction, page 30-56](#)
- [Configuring Integrity Server Support, page 30-57](#)

Overview of Integrity Server and Security Appliance 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, security appliance, 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 security appliance and tells the security appliance what type of firewall client it is.
2. Once it approves the client firewall type, the security appliance passes Integrity server address information back to the Integrity client.
3. With the security appliance 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 server.
4. The Integrity server determines if the Integrity client is in compliance with the mandated security policies. If the client is in compliance with security policies, the Integrity server instructs the security appliance to open the connection and provide the 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 client can no enter the private network.
6. Once the connection is established, the server continues to monitor the state of the client using client heartbeat messages.

**Note**

The current release of the security appliance supports one Integrity Server at a time even though the user interfaces support the configuration of up to five Integrity Servers. If the active Server fails, configure another Integrity Server on the security appliance and then reestablish the client VPN session.

Configuring Integrity Server Support

This section describes an example procedure for configuring the security appliance to support the Zone Labs Integrity Servers. The procedure involves configuring address, port, connection fail timeout and fail states, and SSL certificate parameters.

First, you must configure the hostname or IP address of the Integrity server. The following example commands, entered in global configuration mode, configure an Integrity server using the IP address 10.0.0.5. They also specify port 300 (the default port is 5054) and the inside interface for communications with the Integrity server.

```
hostname(config)# zonelabs-integrity server-address 10.0.0.5
hostname(config)# zonelabs-integrity port 300
hostname(config)# zonelabs-integrity interface inside
hostname(config)#
```

If the connection between the security appliance 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. The following commands ensure that the security appliance waits 12 seconds for a response from either the active or standby Integrity servers before declaring an the Integrity server as failed and closing the VPN client connections:

```
hostname(config)# zonelabs-integrity fail-timeout 12
hostname(config)# zonelabs-integrity fail-close
hostname(config)#
```

The following command returns the configured VPN client connection fail state to the default and ensures the client connections remain open:

```
hostname(config)# zonelabs-integrity fail-open
hostname(config)#
```

The following example commands specify that the Integrity server connects to port 300 (default is port 80) on the security appliance to request the server SSL certificate. While the server SSL certificate is always authenticated, these commands also specify that the client SSL certificate of the Integrity server be authenticated.

```
hostname(config)# zonelabs-integrity ssl-certificate-port 300
hostname(config)# zonelabs-integrity ssl-client-authentication
hostname(config)#
```

To set the firewall client type to the Zone Labs Integrity type, use the **client-firewall** command as described in the “[Configuring Firewall Policies](#)” section on page 30-55. The command arguments that specify firewall policies are not used when the firewall type is **zonelabs-integrity** because the Integrity server determines the policies.

Setting Up Client Firewall Parameters

Enter the following commands to set the appropriate client firewall parameters. You can configure only one instance of each command. [Table 30-2](#) lists the syntax elements of these commands:

Cisco Integrated Firewall

```
hostname(config-group-policy)# client-firewall {opt | req} cisco-integrated acl-in ACL
acl-out ACL
```

Cisco Security Agent

```
hostname(config-group-policy)# client-firewall {opt | req} cisco-security-agent
```

No Firewall

```
hostname(config-group-policy)# client-firewall none
```

Custom Firewall

```
hostname(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

```
hostname(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.

```
hostname(config-group-policy)# client-firewall {opt | req} zonelabs-zonealarm policy {AYT
| CPP acl-in ACL acl-out ACL}
```

```
hostname(config-group-policy)# client-firewall {opt | req} zonelabs-zonealarmpro 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

```
hostname(config-group-policy)# client-firewall {opt | req} sygate-personal
```

```
hostname(config-group-policy)# client-firewall {opt | req} sygate-personal-pro
```

```
hostname(config-group-policy)# client-firewall {opt | req} sygate-security-agent
```

Network Ice, Black Ice Firewall:

```
hostname(config-group-policy)# client-firewall {opt | req} networkkice-blackice
```

Table 30-2 *client-firewall Command Keywords and Variables*

Parameter	Description
acl-in <i>ACL</i>	Provides the policy the client uses for inbound traffic.
acl-out <i>ACL</i>	Provides the policy the client uses for outbound traffic.
AYT	Specifies that the client PC firewall application controls the firewall policy. The security appliance checks to make sure that the firewall is running. It asks, “Are You There?” If there is no response, the security appliance 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 <i>string</i>	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 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:

```
hostname(config)# group-policy FirstGroup attributes
hostname(config-group-policy)# client-firewall req cisco-security-agent
hostname(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 security appliance 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 security appliance permits all connection types.
- When a client matches none of the rules, the security appliance denies the connection. If you define a deny rule, you must also define at least one permit rule; otherwise, the security appliance 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 rule 3 deny type * version 3.*** creates a priority 3 client access rule that denies all client types running release 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:

```
hostname(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.

```
hostname(config-group-policy)# client-access rule priority {permit | deny} type type version {version | none}
```

```
hostname(config-group-policy)# no client-access rule [priority {permit | deny} type type version version]
```

Table 30-3 explains the meaning of the keywords and parameters in these commands.

Table 30-3 *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.
<i>priority</i>	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 security appliance ignores it.

Table 30-3 *client-access rule Command Keywords and Variables*

type <i>type</i>	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 <i>version</i>	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:

```
hostname(config)# group-policy FirstGroup attributes
hostname(config-group-policy)# client-access-rule 1 deny type WinNT version *
hostname(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 security appliance at connect time.

Configuring Group-Policy WebVPN Attributes

WebVPN lets users establish a secure, remote-access VPN tunnel to the security appliance using a web browser. There is no need for either a software or hardware client. WebVPN provides easy access to a broad range of web resources and web-enabled applications from almost any computer that can reach HTTPS Internet sites. WebVPN 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 security appliance recognizes connections that need to be proxied, and the HTTP server interacts with the authentication subsystem to authenticate users. By default, WebVPN is disabled.

You can customize a WebVPN configuration for specific internal group policies.



Note

The webvpn mode that you enter from global configuration mode lets you configure global settings for WebVPN. The webvpn mode described in this section, which you enter from group-policy configuration mode, lets you customize a WebVPN configuration for specific group policies.

In group-policy webvpn configuration mode, you can specify whether to inherit the settings for all the functions or customize the following parameters, each of which is described in the subsequent sections:

- functions
- customizations
- html-content-filter
- homepage
- filter
- url-list
- port-forward
- port-forward-name
- sso server (single-signon server)

- auto-signon
- deny message
- SSL VPN Client (SVC)
- keep-alive ignore
- HTTP compression

In many instances, you define the WebVPN attributes as part of configuring WebVPN, 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, MAPI proxy, URLs and TCP applications over WebVPN. They also identify ACLs and types of traffic to filter. WebVPN is disabled by default. See the description of WebVPN in *Cisco Security Appliance Command Line Configuration Guide* and *Cisco Security Appliance Command Reference* for more information about configuring the WebVPN attributes.

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.

```
hostname(config-group-policy)# webvpn
hostname(config-group-policy)# no webvpn
```

You do not need to configure WebVPN to use e-mail proxies.

The following example shows how to enter group-policy webvpn configuration mode for the group policy named FirstGroup:

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

Configuring Group-Policy WebVPN Function Attributes

Configure the WebVPN functions that you want to enable. To configure file access and file browsing, HTTP Proxy, MAPI Proxy, and URL entry over WebVPN for this group policy, enter the **functions** command in webvpn mode. These functions are disabled by default.

```
hostname(config-group-webvpn)# functions {auto-download | citrix | file-access |
file-browsing | file-entry | filter | http-proxy | mapi | none | port-forward | url-entry}
```

To remove a configured function, enter the **no** form of this command. The **no** option allows inheritance of a value from another group policy. To prevent inheriting function values, enter the **functions none** command.

To remove all configured functions, including a null value created by issuing the **functions none** command, enter the **no** form of this command without arguments.

```
hostname(config-group-webvpn)# no functions [auto-download | citrix | file-access |
file-browsing | file-entry | filter | http-proxy | mapi | none | port-forward | url-entry]
```

Table 30-4 describes the meaning of the keywords used in this command.

Table 30-4 functions Command Keywords

Keyword	Meaning
auto-download	Automatically downloads the port forwarding applet

Table 30-4 *functions Command Keywords*

citrix	Enables Citrix
file-access	Enables or disables file access. When enabled, the WebVPN home page lists file servers in the server list. You must enable file access to enable file browsing and/or file entry.
file-browsing	Enables or disables browsing for file servers and shares. You must enable file browsing to allow user entry of a file server.
file-entry	Enables or disables user ability to enter names of file servers.
filter	Applies a webtype access-list
http-proxy	Enables or disables the forwarding of an HTTP applet proxy to the client. The proxy is useful for technologies that interfere with proper mangling, such as Java, ActiveX, and Flash. It bypasses mangling while ensuring the continued use of the security appliance. The forwarded proxy modifies the browser's old proxy configuration automatically and redirects all HTTP and HTTPS requests to the new proxy configuration. It supports virtually all client side technologies, including HTML, CSS, JavaScript, VBScript, ActiveX, and Java. The only browser it supports is Microsoft Internet Explorer.
mapi	Enables or disables Microsoft Outlook/Exchange port forwarding.
none	Sets a null value for all WebVPN functions . Prevents inheriting functions from a default or specified group policy
port-forward	Enables port forwarding
url-entry	Enables or disables user entry of URLs. When enabled, the security appliance still restricts URLs with any configured URL or network ACLs. When URL entry is disabled, the security appliance restricts WebVPN users to the URLs on the home page.

The following example shows how to configure file access, file browsing, and MAPI Proxy for the group policy named FirstGroup:

```
hostname(config)# group-policy FirstGroup attributes
hostname(config-group-policy)# webvpn
hostname(config-group-webvpn)# functions file-access file-browsing MAPI
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 WebVPN. 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:

```
hostname(config-group-webvpn)# customization customization_name
hostname(config-group-webvpn)#
```

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

```
hostname(config-group-webvpn)# customization blueborder
hostname(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 WebVPN customization named 123 that defines a password prompt. The example then defines a WebVPN group policy named testpolicy and uses the **customization** command to specify the use of the WebVPN customization named 123:

```
hostname(config)# webvpn
hostname(config-webvpn)# customization 123
hostname(config-webvpn-custom)# password-prompt Enter password
hostname(config-webvpn)# exit
hostname(config)# group-policy testpolicy nopassword
hostname(config)# group-policy testpolicy attributes
hostname(config-group-policy)# webvpn
hostname(config-group-webvpn)# customization value 123
hostname(config-group-webvpn)#
```

Specifying a “Deny” Message

You can specify the message delivered to a remote user who logs into WebVPN successfully, but has no VPN privileges by entering the **deny-message** command in group-policy webvpn configuration mode:

```
hostname(config-group-webvpn)# deny-message value "message"
hostname(config-group-webvpn)# no deny-message value "message"
hostname(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 tunnel group 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.

```
hostname(config)# group-policy group2 internal
hostname(config)# group-policy group2 attributes
hostname(config-group)# webvpn
hostname(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."
hostname(config-group-webvpn)
```

Configuring Group-Policy WebVPN Filter Attributes

Specify whether to filter Java, ActiveX, images, scripts, and cookies for WebVPN 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 30-5 describes the meaning of the keywords used in this command.

Table 30-5 filter Command Keywords

Keyword	Meaning
cookies	Removes cookies from images, providing limited ad filtering and privacy.
images	Removes references to images (removes 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:

```
hostname(config)# group-policy FirstGroup attributes
hostname(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 WebVPN home page. 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 WebVPN users. It passes the WebVPN 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 WebVPN users ExamplePolicy group policy, using either basic or NTLM authentication, to servers defined by the URI mask `https://*.example.com/*`:

```
hostname(config)# group-policy ExamplePolicy attributes
hostname(config-group-policy)# webvpn
hostname(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 WebVPN users ExamplePolicy2 group policy, using either basic or NTLM authentication, to the server with the IP address 10.1.1.0, using subnet mask 255.255.255.0:

```
hostname(config)# group-policy ExamplePolicy attributes
hostname(config-group-policy)# webvpn
hostname(config-group-webvpn)# auto-signon allow ip 10.1.1.0 255.255.255.0 auth-type all
hostname(config-group-webvpn)#
```

Specifying the Access List to Use for WebVPN Connections

Specify the name of the access list to use for WebVPN connections for this group policy or username by using the **filter** command in webvpn mode. WebVPN access lists do not apply until you enter the **filter** command to specify them.

To remove the access list, 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.

WebVPN access lists 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 WebVPN traffic.

```
hostname(config-group-webvpn)# filter {value ACLname | none}
hostname(config-group-webvpn)# no filter
```

The **none** keyword indicates that there is no **webvpntype** access list. It sets a null value, thereby disallowing an access list and prevents inheriting an access list from another group policy.

The *ACLname* string following the keyword **value** provides the name of the previously configured access list.

**Note**

You must have **filter** selected in the **functions** command. WebVPN does not use ACLs defined in the **vpn-filter** command.

The following example shows how to set a filter that invokes an access list named *acl_in* for the group policy named FirstGroup:

```
hostname(config)# group-policy FirstGroup attributes
hostname(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 WebVPN 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 WebVPN servers and URLs 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 30-6 shows the **url-list** command parameters and their meanings.

Table 30-6 *url-list Command Keywords and Variables*

Parameter	Meaning
<i>index</i>	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.
<i>value name</i>	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:

```
hostname(config)# group-policy FirstGroup attributes
hostname(config-group-policy)# webvpn
hostname(config-group-webvpn)# url-list value FirstGroupURLs 1
hostname(config-group-webvpn)#
```

Enabling WebVPN Application Access for a Group Policy

To enable WebVPN 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 WebVPN connection. 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 WebVPN users 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*:

```
hostname(config)# group-policy FirstGroup internal attributes
hostname(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*:

```
hostname(config)# group-policy FirstGroup internal attributes
hostname(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 security appliance 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:

```
hostname(config-group-webvpn)# keep-alive-ignore size
hostname(config-group-webvpn)#
```

The **no** form of the command removes this specification from the configuration:

```
hostname(config-group-webvpn)# no keep-alive-ignore
hostname(config-group-webvpn)#
```

The following example sets the maximum size of objects to ignore as 5 KB:

```
hostname(config-group-webvpn)# keep-alive-ignore 5
hostname(config-group-webvpn)#
```

Specifying HTTP Compression

Enable compression of http data over a WebVPN connection for a specific group or user by entering the **http-comp** command in the group policy webvpn mode.

```
hostname(config-group-webvpn)# http-comp {gzip | none}
hostname(config-group-webvpn)#
```

To remove the command from the configuration and cause the value to be inherited, use the **no** form of the command:

```
hostname(config-group-webvpn)# no http-comp {gzip | none}
hostname(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 WebVPN connections, 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 WebVPN, 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.

```
hostname(config-group-webvpn)# sso-server value server_name
hostname(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.

```
hostname(config-group-webvpn)# sso-server {value server_name | none}
hostname(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”:

```
hostname(config)# group-policy my-sso-grp-pol internal
hostname(config)# group-policy my-sso-grp-pol attributes
hostname(config-group-policy)# webvpn
hostname(config-group-webvpn)# sso-server value example
hostname(config-group-webvpn)#
```

Configuring SVC

The SSL VPN Client (SVC) is a VPN tunneling technology that gives remote users the benefits of an IPsec VPN client without the need for network administrators to install and configure IPsec VPN clients on remote computers. The SVC uses the SSL encryption that is already present on the remote computer as well as the WebVPN login and authentication of the security appliance.

To establish an SVC session, the remote user enters the IP address of a WebVPN interface of the security appliance in the browser, and the browser connects to that interface and displays the WebVPN login screen. If the user satisfies the login and authentication, and the security appliance identifies the user as *requiring* the SVC, the security appliance downloads the SVC to the remote computer. If the security appliance identifies the user as having the *option* to use the SVC, the security appliance downloads the SVC to the remote computer while presenting a link on the user screen to skip the SVC installation.

After downloading, the SVC installs and configures itself, and then the SVC either remains or uninstalls itself (depending on the configuration) from the remote computer when the connection terminates.

The security appliance might have several unique SVC images residing in cache memory for different remote computer operating systems. When the user attempts to connect, the security appliance can consecutively download portions of these images to the remote computer until the image and operating system match, at which point it downloads the entire SVC. You can order the SVC images to minimize connection setup time, with the first image downloaded representing the most commonly-encountered remote computer operating system. For complete information about installing and using SVC, see [Chapter 38, “Configuring SSL VPN Client”](#).

After enabling SVC, as described in [Chapter 38, “Configuring SSL VPN Client”](#), you can enable or require SVC features for a specific group. This feature is disabled by default. If you enable or require SVC, you can then enable a succession of `svc` commands, described in this section. To enable SVC and its related `svc` commands, do the following steps in `group-policy webvpn` configuration mode:

- Step 1** To enable the security appliance to download SVC files to remote computers, enter the **svc enable** command. By default, this command is disabled. The security appliance does not download SVC files. To remove the **svc enable** command from the configuration, use the **no** form of this command.

```
hostname(config-group-webvpn)# svc {none | enable | required}
hostname(config-group-webvpn)#
```



Note

Entering the **no svc enable** command does not terminate active SVC sessions.

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

- Step 2** To enable compression of HTTP data over an SVC connection, for a specific group, enter the `svc compression` command. By default, SVC compression is set to **deflate** (enabled). To disable compression for a specific group, use the **none** keyword. To remove the `svc compression` command and cause the value to be inherited, use the **no** form of the command:

```
hostname(config-group-webvpn)# svc compression {deflate | none}
hostname(config-group-webvpn)#
```

The following example disables SVC compression for the group policy named `sales`:

```
hostname(config)# group-policy sales attributes
hostname(config-group-policy)# webvpn
hostname(config-group-webvpn)# svc compression none
hostname(config-group-webvpn)#
```

- Step 3** To enable dead-peer-detection (DPD) on the security appliance and to set the frequency with which either the SVC or the security appliance performs DPD, use the **svc dpd-interval** command. To remove the **svc dpd-interval** command from the configuration, use the **no** form of the command. To disable SVC DPD for this group, use the **none** keyword:

```
hostname(config-group-webvpn)# svc dpd-interval {[gateway {seconds | none}] | [client
{seconds | none}]}
hostname(config-group-webvpn)#
```

DPD checking is disabled by default.

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

The **client** refers to the SVC. 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.

In the following example, the user configures the DPD frequency performed by the security appliance (**gateway**) to 3000 seconds, and the DPD frequency performed by the client to 1000 seconds for the existing group policy named **sales**:

```
hostname(config)# group-policy sales attributes
hostname(config-group-policy)# webvpn
hostname(config-group-webvpn)# svc dpd-interval gateway 3000
hostname(config-group-webvpn)# svc dpd-interval client 1000
hostname(config-group-webvpn)#
```

- Step 4** You can adjust the frequency of keepalive messages (specified by *seconds*), to ensure that an SVC connection through a proxy, firewall, or NAT device remains open, even if the device limits the time that the connection can be idle.

Adjusting the frequency also ensures that the SVC 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.

To configure the frequency (15 through 600 seconds) which an SVC on a remote computer sends keepalive messages to the security appliance, use the **svc keepalive** command. Use the **no** form of the command to remove the command from the configuration and cause the value to be inherited:

```
hostname(config-group-webvpn)# svc keepalive {none | seconds}
hostname(config-group-webvpn)# no svc keepalive {none | seconds}
hostname(config-group-webvpn)#
```

SVC keepalives are disabled by default. Using the keyword **none** disables SVC keepalive messages.

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

```
hostname(config-group-webvpn)# svc keepalive 300
hostname(config-group-webvpn)#
```

- Step 5** To enable the permanent installation of an SVC onto a remote computer, use the **svc keep-installer** command with the **installed** keyword. To remove the command from the configuration, use the **no** form of this command:

```
hostname(config-group-webvpn)# svc keep-installer {installed | none}
hostname(config-group-webvpn)# no svc keep-installer {installed | none}
hostname(config-group-webvpn)#
```

The default is that permanent installation of the SVC is disabled. The SVC uninstalls from the remote computer at the end of the SVC session.

The following example configures the security appliance to keep the SVC installed on the remote computer for this group:

```
hostname(config-group-webvpn)# svc keep-installer installed
hostname(config-group-webvpn)#
```

Step 6 To enable the SVC to perform a rekey on an SVC session, use the **svc rekey** command. To disable rekey and remove the command from the configuration, use the **no** form of this command:

```
hostname(config-group-webvpn)# svc rekey {method {ssl | new-tunnel} | time minutes | none}
hostname(config-group-webvpn)# no svc rekey {method {ssl | new-tunnel} | time minutes | none}
hostname(config-group-webvpn)#
```

By default, SVC rekey is disabled.

Specifying the method as `new-tunnel` specifies that the SVC establishes a new tunnel during SVC rekey. Specifying the method as `none` disables SVC rekey. Specifying the method as `ssl` specifies that SSL renegotiation takes place during SVC rekey. 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).

For the **no** form of the command, only the minimum is necessary, as the following example shows:

```
hostname(config-username-webvpn)# no svc rekey method
hostname(config-username-webvpn)#
```

If, however, you specify the method as `new-tunnel`:

```
hostname(config-username-webvpn)# no svc rekey method new-tunnel
hostname(config-username-webvpn)#
```

but the current method is `ssl`, then the command fails, because the values don't match.

In the following example, the user configures the SVC to renegotiate with SSL during rekey and configures the rekey to occur 30 minutes after the session begins:

```
hostname(config-group-webvpn)# svc rekey method ssl
hostname(config-group-webvpn)# svc rekey time 30
hostname(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 30-73](#)
- [Configuring Attributes for Specific Users, page 30-73](#)

By default, users inherit all user attributes from the assigned group policy. The security appliance 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:

```
hostname# show running-config all username
hostname#
```

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:

```
hostname# show running-config all username testuser
username testuser password 12RsxXQnphyr/I9Z encrypted privilege 15
```

Configuring Attributes for Specific 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 security appliance 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.

```
hostname(config)# username name {nopassword | password password [encrypted]} [privilege
priv_level]}
```

```
hostname(config)# no username [name]
```

[Table 30-7](#) describes the meaning of the keywords and variables used in this command.

Table 30-7 *username Command Keywords and Variables*

Keyword/Variable	Meaning
encrypted	Indicates that the password is encrypted.
<i>name</i>	Provides the name of the user.
nopassword	Indicates that this user needs no password.

password <i>password</i>	Indicates that this user has a password, and provides the password.
privilege <i>priv_level</i>	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 security appliance. 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:

```
hostname(config)# username anyuser password pw_12345678 encrypted privilege 12
hostname(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:

```
hostname(config)# username name attributes
hostname(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:

```
hostname(config-username)# vpn-group-policy group-policy-name
hostname(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:

```
hostname(config)# username anyuser attributes
hostname(config-username)# vpn-group-policy FirstGroup
hostname(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.

```
hostname(config-username)# vpn-access-hours value {time-range | none}
hostname(config-username)# vpn-access-hours value none
hostname(config)#
```

The following example shows how to associate the user named anyuser with a time-range policy called 824:

```
hostname(config)# username anyuser attributes
hostname(config-username)# vpn-access-hours 824
hostname(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.

```
hostname(config-username)# vpn-simultaneous-logins integer
hostname(config-username)# no vpn-simultaneous-logins
hostname(config-username)#
```



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:

```
hostname(config)# username anyuser attributes
hostname(config-username)# vpn-simultaneous-logins 4
hostname(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 security appliance terminates the connection.

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.

```
hostname(config-username)# vpn-idle-timeout {minutes | none}
hostname(config-username)# no vpn-idle-timeout
hostname(config-username)#
```

The following example shows how to set a VPN idle timeout of 15 minutes for the user named anyuser:

```
hostname(config)# username anyuser attributes
hostname(config-username)# vpn-idle-timeout 30
hostname(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 security appliance terminates the connection.

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.

```
hostname(config-username)# vpn-session-timeout {minutes | none}
hostname(config-username)# no vpn-session-timeout
hostname(config-username)#
```

The following example shows how to set a VPN session timeout of 180 minutes for the user named anyuser:

```
hostname(config)# username anyuser attributes
hostname(config-username)# vpn-session-timeout 180
hostname(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 access list and prevent inheriting an access list 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.

```
hostname(config-username)# vpn-filter {value ACL_name | none}
hostname(config-username)# no vpn-filter
hostname(config-username)#
```



Note

You must have **filter** selected in the **functions** command. WebVPN does not use ACLs defined in the **vpn-filter** command.

The following example shows how to set a filter that invokes an access list named `acl_vpn` for the user named anyuser:

```
hostname(config)# username anyuser attributes
hostname(config-username)# vpn-filter value acl_vpn
hostname(config-username)#
```

Specifying the IP 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.

```
hostname(config-username)# vpn-framed-ip-address {ip_address}
hostname(config-username)# no vpn-framed-ip-address
hostname(config-username)
```

The following example shows how to set an IP address of 10.92.166.7 for a user named anyuser:

```
hostname(config)# username anyuser attributes
hostname(config-username)# vpn-framed-ip-address 10.92.166.7
hostname(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.

```
hostname(config-username)# vpn-framed-ip-netmask {netmask}
hostname(config-username)# no vpn-framed-ip-netmask
hostname(config-username)
```

The following example shows how to set a subnet mask of 255.255.255. 254 for a user named anyuser:

```
hostname(config)# username anyuser attributes
hostname(config-username)# vpn-framed-ip-netmask 255.255.255.254
hostname(config-username)
```

Specifying the Tunnel Protocol

Specify the VPN tunnel types (IPSec or WebVPN) 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.

```
hostname(config-username)# vpn-tunnel-protocol {webvpn | IPSec}
hostname(config-username)# no vpn-tunnel-protocol [webvpn | IPSec]
hostname(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 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 WebVPN and IPSec tunneling modes for the user named anyuser:

```
hostname(config)# username anyuser attributes
hostname(config-username)# vpn-tunnel-protocol webvpn
hostname(config-username)# vpn-tunnel-protocol IPSec
hostname(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 tunnel group. Group-lock restricts users by checking whether the group configured in the VPN client is the same as the tunnel group to which the user is assigned. If it is not, the security appliance prevents the user from connecting. If you do not configure group-lock, the security appliance 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.

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

The following example shows how to set group lock for the user named anyuser:

```
hostname(config)# username anyuser attributes
hostname(config-username)# group-lock value tunnel-group-name
hostname(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.

```
hostname(config-username)# password-storage {enable | disable}
hostname(config-username)# no password-storage
hostname(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:

```
hostname(config)# username anyuser attributes
hostname(config-username)# password-storage enable
hostname(config-username)
```

Configuring WebVPN for Specific Users

The following sections describe how to customize a WebVPN configuration for specific users. Enter username webvpn configuration mode by using the **webvpn** command in username configuration mode. WebVPN lets users establish a secure, remote-access VPN tunnel to the security appliance using a web browser. There is no need for either a software or hardware client. WebVPN provides easy access to a broad range of web resources and web-enabled applications from almost any computer that can reach HTTPS Internet sites. WebVPN 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 security appliance 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, MAPI proxy, URLs and TCP applications over WebVPN. They also identify ACLs and types of traffic to filter. WebVPN 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.

```
hostname(config-username)# webvpn
hostname(config-username-webvpn)#
```

To remove all commands entered in username webvpn configuration mode, use the **no** form of this command:

```
hostname(config-username)# no webvpn
hostname(config-username)#
```

You do not need to configure WebVPN to use e-mail proxies.



Note

The `webvpn` mode that you enter from global configuration mode lets you configure global settings for WebVPN. The username `webvpn` configuration mode described in this section, which you enter from username mode, lets you customize a WebVPN configuration for specific users.

In username `webvpn` configuration mode, you can customize the following parameters, each of which is described in the subsequent steps:

- `functions`
- `customizations`
- `deny message`
- `html-content-filter`
- `homepage`
- `filter`
- `url-list`
- `port-forward`
- `port-forward-name`
- `sso server` (single-signon server)
- `auto-signon`
- `SSL VPN Client (SVC)`
- `keep-alive ignore`
- `HTTP compression`

The following example shows how to enter username `webvpn` configuration mode for the username `anyuser` attributes:

```
hostname(config)# username anyuser attributes
hostname(config-username)# webvpn
hostname(config-username-webvpn)#
```

Configuring the WebVPN Functions to Enable

To configure the WebVPN functions file access and file browsing, HTTP Proxy, MAPI Proxy, and URL entry over WebVPN for this user, enter the **functions** command in username `webvpn` configuration mode. To remove a configured function, enter the **no** form of this command. These functions are disabled by default.

To remove all configured functions, including a null value created by issuing the **functions none** command, enter the **no** form of this command without arguments. The **no** option allows inheritance of a value from another group policy. To prevent inheriting function values, enter the **functions none** command.

```
hostname(config-username-webvpn)# functions {auto-download | citrix | file-access |
file-browsing | file-entry | filter | http-proxy | mapi | none | port-forward | url-entry}

hostname(config-username-webvpn)# no functions [auto-download | citrix | file-access |
file-browsing | file-entry | filter | http-proxy | mapi | none | port-forward | url-entry]
```

Table 30-8 describes the meaning of the keywords used in this command.

Table 30-8 *functions Command Keywords*

Keyword	Meaning
auto-download	Automatically downloads the port forwarding applet
citrix	Enables Citrix
file-access	Enables or disables file access. When enabled, the WebVPN home page lists file servers in the server list. You must enable file access to enable file browsing and/or file entry.
file-browsing	Enables or disables browsing for file servers and shares. You must enable file browsing to allow user entry of a file server.
file-entry	Enables or disables user ability to enter names of file servers.
filter	Applies a webtype access-list
http-proxy	Enables or disables the forwarding of an HTTP applet proxy to the client. The proxy is useful for technologies that interfere with proper mangling, such as Java, ActiveX, and Flash. It bypasses mangling while ensuring the continued use of the security appliance. The forwarded proxy modifies the browser's old proxy configuration automatically and redirects all HTTP and HTTPS requests to the new proxy configuration. It supports virtually all client side technologies, including HTML, CSS, JavaScript, VBScript, ActiveX, and Java. The only browser it supports is Microsoft Internet Explorer.
mapi	Enables or disables Microsoft Outlook/Exchange port forwarding.
none	Sets a null value for all WebVPN functions . Prevents inheriting functions from a default or specified group policy
port-forward	Enables port forwarding
url-entry	Enables or disables user entry of URLs. When enabled, the security appliance still restricts URLs with any configured URL or network ACLs. When URL entry is disabled, the security appliance restricts WebVPN users to the URLs on the home page.

The keywords used in this command are as follows:

- **file-access**—Enables or disables file access. When enabled, the WebVPN home page lists file servers in the server list. You must enable file access to enable file browsing and/or file entry.
- **file-browsing**—Enables or disables browsing for file servers and shares. You must enable file browsing to allow user entry of a file server.
- **file-entry**—Enables or disables user ability to enter names of file servers.
- **http-proxy**—Enables or disables the forwarding of an HTTP applet proxy to the client. The proxy is useful for technologies that interfere with proper mangling, such as Java, ActiveX, and Flash. It bypasses mangling while ensuring the continued use of the security appliance. The forwarded proxy modifies the browser's old proxy configuration automatically and redirects all HTTP and HTTPS requests to the new proxy configuration. It supports virtually all client side technologies, including HTML, CSS, JavaScript, VBScript, ActiveX, and Java. The only browser it supports is Microsoft Internet Explorer.
- **mapi**—Enables or disables Microsoft Outlook/Exchange port forwarding.

- **none**—Sets a null value for all WebVPN **functions**. Prevents inheriting functions from a default or specified group policy
- **url-entry**—Enables or disables user entry of URLs. When enabled, the security appliance still restricts URLs with any configured URL or network ACLs. When URL entry is disabled, the security appliance restricts WebVPN users to the URLs on the home page.

The following example shows how to configure file access, file browsing, HTTP Proxy, and MAPI Proxy for the user named anyuser:

```
hostname(config)# username anyuser attributes
hostname(config-username)# webvpn
hostname(config-username-webvpn)# functions file-access file-browsing MAPI
hostname(config-username-webvpn)#
```

Specifying the Content/Objects to Filter from the HTML

To filter Java, ActiveX, images, scripts, and cookies for WebVPN 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 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:

```
hostname(config)# username anyuser attributes
hostname(config-username)# webvpn
hostname(config-username-webvpn)# html-content-filter java cookies images
hostname(config-username-webvpn)#
```

Specifying the User Home Page

To specify a URL for the web page that displays when this WebVPN user logs in, 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 WebVPN 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
hostname(config-username-webvpn)#
```

The following example shows how to specify `www.example.com` as the home page for the user named `anyuser`:

```
hostname(config)# username anyuser attributes
hostname(config-username)# webvpn
hostname(config-username-webvpn)# homepage value www.example.com
hostname(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 WebVPN. 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:

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

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

```
hostname(config-username-webvpn)# customization value blueborder
hostname(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 WebVPN customization named `123` that defines a password prompt. The example then defines a WebVPN tunnel-group named `test` and uses the **customization** command to specify the use of the WebVPN customization named `123`:

```
hostname(config)# webvpn
hostname(config-webvpn)# customization 123
hostname(config-webvpn-custom)# password-prompt Enter password
hostname(config-webvpn)# exit
hostname(config)# username testuser nopassword
hostname(config)# username testuser attributes
hostname(config-username-webvpn)# webvpn
hostname(config-username-webvpn)# customization value 123
hostname(config-username-webvpn)#
```

Specifying a “Deny” Message

You can specify the message delivered to a remote user who logs into WebVPN successfully, but has no VPN privileges by entering the **deny-message** command in username webvpn configuration mode:

```
hostname(config-username-webvpn)# deny-message value "message"
hostname(config-username-webvpn)# no deny-message value "message"
hostname(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 tunnel group 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.

```
hostname(config)# username anyuser attributes
hostname(config-username)# webvpn
hostname(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."
hostname(config-username-webvpn)
```

Specifying the Access List to use for WebVPN Connections

To specify the name of the access list to use for WebVPN connections for this user, enter the **filter** command in username webvpn configuration mode. To remove the access list, 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.

WebVPN access lists 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 WebVPN traffic.

```
hostname(config-username-webvpn)# filter {value ACLname | none}
hostname(config-username-webvpn)# no filter
hostname(config-username-webvpn)#
```

The **none** keyword indicates that there is no **webvpntype** access list. It sets a null value, thereby disallowing an access list and prevents inheriting an access list from another group policy.

The *ACLname* string following the keyword **value** provides the name of the previously configured access list.



Note

WebVPN does not use ACLs defined in the **vpn-filter** command.

The following example shows how to set a filter that invokes an access list named *acl_in* for the user named anyuser:

```
hostname(config)# username anyuser attributes
hostname(config-username)# webvpn
hostname(config-username-webvpn)# filter acl_in
hostname(config-username-webvpn)#
```

Applying a URL List

You can specify a list of URLs to appear on the WebVPN home page for a user. First, you must create one or more named lists by entering the **url-list** command in global configuration mode. To apply a list of WebVPN servers and URLs to a particular user, 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 WebVPN end user interface.
- *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 WebVPN users 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:

```
hostname(config)# username anyuser attributes
hostname(config-username)# webvpn
hostname(config-username-webvpn)# url-list value AnyuserURLs
hostname(config-username-webvpn)#
```

Enabling WebVPN Application Access

To enable WebVPN 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
hostname(config-username-webvpn)#
```

The *listname* string following the keyword **value** identifies the list of applications WebVPN users 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 WebVPN connection. 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:

```
hostname(config-group-policy)# webvpn
hostname(config-username-webvpn)# port-forward value ports1
hostname(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:

```
hostname(config-group-policy)# webvpn
hostname(config-username-webvpn)# port-forward-name value test
hostname(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 security appliance 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:

```
hostname(config-group-webvpn)# keep-alive-ignore size
hostname(config-group-webvpn)#
```

The **no** form of the command removes this specification from the configuration:

```
hostname(config-group-webvpn)# no keep-alive-ignore
hostname(config-group-webvpn)#
```

The following example sets the maximum size of objects to ignore as 5 KB:

```
hostname(config-group-webvpn)# keep-alive-ignore 5
hostname(config-group-webvpn)#
```

Configuring Auto-Signon

To automatically submit the WebVPN login credentials of a particular WebVPN user 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 WebVPN users. It passes the WebVPN 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 WebVPN user named anyuser, using either basic or NTLM authentication, to servers defined by the URI mask `https://*.example.com/*`:

```
hostname(config)# username anyuser attributes
hostname(config-username)# webvpn
hostname(config-username-webvpn)# auto-signon allow uri https://*.example.com/* auth-type
all
```

The following example commands configure auto-signon for the WebVPN user 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:

```
hostname(config)# username anyuser attributes
hostname(config-username)# webvpn
hostname(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 WebVPN connection for a specific user by entering the **http-comp** command in the username webvpn configuration mode.

```
hostname(config-username-webvpn)# http-comp {gzip | none}
hostname(config-username-webvpn)#
```

To remove the command from the configuration and cause the value to be inherited, use the **no** form of the command:

```
hostname(config-username-webvpn)# no http-comp {gzip | none}
hostname(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 WebVPN connections, 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 WebVPN, 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.

```
hostname(config-username-webvpn)# sso-server value server_name
hostname(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.

```
hostname(config-username-webvpn)# sso-server {value server_name | none}
hostname(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`:

```
hostname(config)# username anyuser attributes
hostname(config-username)# webvpn
hostname(config-username-webvpn)# sso-server value example
hostname(config-username-webvpn)#
```

Configuring SVC

The SSL VPN Client (SVC) is a VPN tunneling technology that gives remote users the benefits of an IPSec VPN client without the need for network administrators to install and configure IPSec VPN clients on remote computers. The SVC uses the SSL encryption that is already present on the remote computer as well as the WebVPN login and authentication of the security appliance.

To establish an SVC session, the remote user enters the IP address of a WebVPN interface of the security appliance in the browser, and the browser connects to that interface and displays the WebVPN login screen. If the user satisfies the login and authentication, and the security appliance identifies the user as *requiring* the SVC, the security appliance downloads the SVC to the remote computer. If the security appliance identifies the user as having the *option* to use the SVC, the security appliance downloads the SVC to the remote computer while presenting a link on the user screen to skip the SVC installation.

After downloading, the SVC installs and configures itself, and then the SVC either remains or uninstalls itself (depending on the configuration) from the remote computer when the connection terminates.

The security appliance might have several unique SVC images residing in cache memory for different remote computer operating systems. When the user attempts to connect, the security appliance can consecutively download portions of these images to the remote computer until the image and operating system match, at which point it downloads the entire SVC. You can order the SVC images to minimize connection setup time, with the first image downloaded representing the most commonly-encountered remote computer operating system. For complete information about installing and using SVC, see [Chapter 38, “Configuring SSL VPN Client”](#).

After enabling SVC, as described in [Chapter 38, “Configuring SSL VPN Client”](#), you can enable or require SVC features for a specific user. This feature is disabled by default. If you enable or require SVC, you can then enable a succession of `svc` commands, described in this section. To enable SVC and its related `svc` commands, do the following steps in `username webvpn` configuration mode:

- Step 1** To enable the security appliance to download SVC files to remote computers, enter the **svc enable** command. By default, this command is disabled. The security appliance does not download SVC files. To remove the **svc enable** command from the configuration, use the **no** form of this command.

```
hostname(config-username-webvpn)# svc {none | enable | required}
hostname(config-username-webvpn)#
```



Note Entering the **no svc enable** command does not terminate active SVC sessions.

```
hostname(config)# username sales attributes
hostname(config-username)# webvpn
hostname(config-username-webvpn)# svc enable
hostname(config-username-webvpn)#
```

- Step 2** To enable compression of HTTP data over an SVC connection, for a specific user, enter the **svc compression** command. By default, SVC compression is set to **deflate** (enabled). To disable compression for a specific user, use the **none** keyword. To remove the **svc compression** command and cause the value to be inherited, use the **no** form of the command:

```
hostname(config-username-webvpn)# svc compression {deflate | none}
hostname(config-username-webvpn)#
```

The following example disables SVC compression for the user named sales:

```
hostname(config)# username sales attributes
hostname(config-username)# webvpn
hostname(config-username-webvpn)# svc compression none
hostname(config-username-webvpn)#
```

- Step 3** To enable dead-peer-detection (DPD) on the security appliance and to set the frequency with which either the SVC or the security appliance performs DPD, use the **svc dpd-interval** command. To remove the **svc dpd-interval** command from the configuration, use the **no** form of the command. To disable SVC DPD for this user, use the **none** keyword:

```
hostname(config-username-webvpn)# svc dpd-interval {[gateway {seconds | none}] | [client
{seconds | none}]}
hostname(config-username-webvpn)#
```

DPD checking is disabled by default.

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

The client refers to the SVC. 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.

In the following example, the user configures the DPD frequency performed by the security appliance (gateway) to 3000 seconds, and the DPD frequency performed by the client to 1000 seconds for the existing user named sales:

```
hostname(config)# username sales attributes
hostname(config-username)# webvpn
hostname(config-username-webvpn)# svc dpd-interval gateway 3000
hostname(config-username-webvpn)# svc dpd-interval client 1000
hostname(config-username-webvpn)#
```


- Step 4** You can adjust the frequency of keepalive messages (specified by *seconds*), to ensure that an SVC connection through a proxy, firewall, or NAT device remains open, even if the device limits the time that the connection can be idle.

Adjusting the frequency also ensures that the SVC 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.

To configure the frequency (15 through 600 seconds) which an SVC on a remote computer sends keepalive messages to the security appliance, use the **svc keepalive** command. Use the **no** form of the command to remove the command from the configuration and cause the value to be inherited:

```
hostname(config-username-webvpn) # svc keepalive {none | seconds}
hostname(config-username-webvpn) # no svc keepalive {none | seconds}
hostname(config-username-webvpn) #
```

SVC keepalives are disabled by default. Using the keyword **none** disables SVC keepalive messages.

In the following example, the user configures the security appliance to enable the SVC to send keepalive messages, with a frequency of 300 seconds (5 minutes):

```
hostname(config-username-webvpn) # svc keepalive 300
hostname(config-username-webvpn) #
```

- Step 5** To enable the permanent installation of an SVC onto a remote computer, use the **svc keep-installer** command with the **installed** keyword. To remove the command from the configuration, use the **no** form of this command:

```
hostname(config-username-webvpn) # svc keep-installer {installed | none}
hostname(config-username-webvpn) # no svc keep-installer {installed | none}
hostname(config-username-webvpn) #
```

The default is that permanent installation of the SVC is disabled. The SVC uninstalls from the remote computer at the end of the SVC session.

The following example configures the security appliance to keep the SVC installed on the remote computer for this user:

```
hostname(config-username-webvpn) # svc keep-installer installed
hostname(config-username-webvpn) #
```

- Step 6** To enable the SVC to perform a rekey on an SVC session, use the **svc rekey** command:

```
hostname(config-username-webvpn) # svc rekey {method {ssl | new-tunnel} | time minutes | none}}
```

To disable rekey and remove the command from the configuration, use the **no** form of this command:

```
hostname(config-username-webvpn) # no svc rekey [method {ssl | new-tunnel} | time minutes | none]
hostname(config-username-webvpn) #
```

By default, SVC rekey is disabled.

Specifying the method as *new-tunnel* specifies that the SVC establishes a new tunnel during SVC rekey. Specifying the method as *none* disables SVC rekey. Specifying the method as *ssl* specifies that SSL renegotiation takes place during SVC rekey. 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).

For the **no** form of the command, only the minimum is necessary. The following example is correct:

```
hostname(config-username-webvpn) # no svc rekey method
hostname(config-username-webvpn) #
```

If, however, you specify the method as new-tunnel:

```
hostname(config-username-webvpn)# no svc rekey method new-tunnel
hostname(config-username-webvpn)#
```

and the current method is ssl, then the command fails, because the values don't match.

In the following example, the user configures the SVC to renegotiate with SSL during rekey and configures the rekey to occur 30 minutes after the session begins:

```
hostname(config-username-webvpn)# svc rekey method ssl
hostname(config-username-webvpn)# svc rekey time 30
hostname(config-username-webvpn)#
```
