

Object Management

This chapter describes how to manage reusable objects.

- Objects, on page 2
- Use the object manager, on page 4
- AAA servers, on page 15
- Access lists, on page 21
- Configure address pools, on page 25
- Create an application filter, on page 26
- Configure an AS PATH object, on page 27
- Configure a BFD template, on page 28
- Cipher suite lists, on page 29
- Configure a community list, on page 30
- DHCP IPv6 pools, on page 33
- Distinguished names, on page 33
- DNS server groups, on page 36
- External attributes, on page 37
- File lists, on page 46
- FlexConfig objects, on page 51
- Geolocation objects, on page 51
- Interfaces, on page 52
- Key chains, on page 53
- Networks, on page 55
- PKI objects, on page 58
- Configure a policy list, on page 78
- Ports, on page 80
- Prefix lists, on page 82
- Configure a route map, on page 84
- Security intelligence, on page 87
- Sinkhole objects, on page 101
- Configure an SLA monitor, on page 102
- Time ranges, on page 104
- Time zones, on page 105
- Tunnel zones, on page 106
- URLs, on page 106

• Variable sets, on page 108
• VLAN tags, on page 125
• VPN, on page 125
• History for object management , on page 145

Objects

An object is a configuration element that

- associates a name with a value for increased flexibility and ease of use in the web interface,
- enables reuse of configurations across policies, rules, event searches, reports, and dashboards, and
- can be managed, grouped, and overridden as needed within the system.

Object management and usage

The system uses named objects to simplify configuration and management. Use the object manager to create and manage objects. Many configurations that use objects also allow you to create objects as needed. The object manager enables you to:

- View the policies, settings, and other objects where a network, port, VLAN, or URL object is used; see View objects and their usage, on page 8.
- Group objects to reference multiple objects with a single configuration; see Object groups, on page 10.
- Override object values for selected devices; see Object overrides, on page 12.

After editing an object used in an active policy, you must redeploy the changed configuration for your changes to take effect. You cannot delete an object that is in use by an active policy.

Note The system configures an object on a managed device only when a policy assigned to that device uses the object. If you remove an object from all policies assigned to a given device, the object is also removed from the device configuration on the next deployment, and subsequent changes to the object are not reflected in the device configuration.

Object types

This table lists the objects you can create in the system, and indicates whether each object type can be grouped or configured to allow overrides.

Object type	Groupable?	Allows overrides?
Network	yes	yes
Port	yes	yes

Object type	Groupable?	Allows overrides?
Interface: • Security Zone • Interface Group	no	no
Tunnel Zone	no	no
Application Filter	no	no
VLAN Tag	yes	yes
External Attribute: Security Group Tag (SGT) and Dynamic Object	no	no
URL	yes	yes
Geolocation	no	no
Time Range	no	no
Variable Set	no	no
Security Intelligence: Network, DNS, and URL lists and feeds	no	no
Sinkhole	no	no
File List	no	no
Cipher Suite List	no	no
Distinguished Name	yes	no
Public Key Infrastructure (PKI): • Internal and Trusted CA • Internal and External Certs	yes	no
Key Chain	no	yes
DNS Server Group	no	no
SLA Monitor	no	no
Prefix List: IPv4 and IPv6	no	yes
Route Map	no	yes
Access List: Standard and Extended	no	yes
AS Path	no	yes
Community List	no	yes

Object type	Groupable?	Allows overrides?
Policy List	no	yes
FlexConfig: Text and FlexConfig objects	no	yes

Objects and multi-tenancy

In a multidomain deployment, you can create objects in Global and descendant domains with the exception of Security Group Tag (SGT) objects, which you can create only in the Global domain. The interface shows objects created in the current domain, which you can edit. It also presents objects created in ancestor domains, which you cannot edit, except for security zones and interface groups.



Note Security zones and interface groups are tied to device interfaces, which you configure at the leaf level. Administrators in descendant domains can view and edit security zones and groups created in ancestor domains. Subdomain users can add and delete interfaces from ancestor zones and groups, but cannot delete or rename these zones or groups.

Object names must be unique within the domain hierarchy. The system may identify a conflict with the name of an object you cannot view in your current domain.

For objects that support grouping, you can group objects in the current domain with objects inherited from ancestor domains.

Object overrides allow you to define device-specific or domain-specific values for certain types of object, including network, port, VLAN tag, and URL. In a multidomain deployment, you can define a default value for an object in an ancestor domain, but allow administrators in descendant domains to add override values for that object.

Use the object manager

The object manager provides capabilities for creating and managing objects and object groups, viewing them in paginated lists, filtering results, and refreshing the displayed information.

- The object manager enables creation and management of objects and object groups.
- The object manager displays 20 objects or groups per page. If you have more than 20 of any type of object or group, use the navigation links at the bottom of the page to view additional pages. You can also go to a specific page or click **Refresh** ( ) to refresh your view.
- Access the page using **Objects** > **Object Management**. You can alternatively access the page using **Objects** > **Other FTD Objects**.
- By default, objects and groups are listed alphabetically by name. You can filter objects by name or value.

Import objects

Import objects into the system by uploading a properly formatted comma-separated values file. This task enables bulk addition of supported object types, streamlining object management.

Objects can be quickly imported into the system using a properly formatted CSV file. Up to 1000 objects can be imported in a single operation. The format and required columns vary depending on object type.

Object TYPE	Rules

Individual object	• The column header must be mentioned in capital letters.
	• The file must have the following columns headers:
	  • NAME
	  • DN
	• Both NAME and DN column entries are mandatory to import an entry.
	• You can import individual objects directly into an existing distinguished NAME object group.

Network object	• The column header must be mentioned in capital letters.
	• The file must have the following columns headers:
	  • NAME
	  • DESCRIPTION
	  • TYPE
	  • VALUE
	  • LOOKUP
	• To import a host, range, or network object type, provide entries for both NAME and VALUE columns.
	• For an FQDN object, specify "FQDN" in the TYPE column entry, and enter "IPv4," "IPv6," or "IPv4/IPv6" in the LOOKUP column entry.
	• If you leave the LOOKUP column entry for the FQDN object empty, the system saves the object with ipv4_ipv6 as the VALUE.

Object TYPE	Rules

PORT	• The column header must be mentioned in capital letters.
	• The file must have the following columns headers:
	  • NAME
	  • PROTOCOL
	  • PORT
	  • ICMPCODE
	  • ICMPTYPE
	• The NAME column entry is mandatory.
	• The PORT column entry is mandatory for 'tcp' and 'udp' PROTOCOL types.
	• The ICMPCODE and ICMPTYPE column entries are mandatory for 'icmp' and 'icmp6' PROTOCOL types.

URL	• The column header must be mentioned in capital letters.
	• The file must have the following columns headers:
	  • NAME
	  • DESCRIPTION
	  • URL
	• You must provide entries for the NAME and URL columns to import an object.

VLAN TAG	• The column header must be mentioned in capital letters.
	• The file must have these columns headers:
	  • NAME
	  • DESCRIPTION
	  • TAG
	• The NAME and TAG column entries are mandatory to import an entry.

Procedure

Step 1 Choose **Objects** > **Object Management**.

Step 2 Choose one of these object types from the left pane:

- **Distinguished Name** > **Individual Objects**
- **Network Object**
- **PORT**
- **URL**
- **VLAN TAG**

Step 3 Choose **Import Object** from the **Add [Object TYPE]** drop-down list.

Note If you have selected **Individual Objects** in the previous step, click **Import**.

Step 4 Click **Browse**.

Step 5 Locate and select the comma-separated file on your system.

Step 6 Click **Open**.

Note While importing **Distinguished Name** objects, you can optionally check the **Add imported Distinguished Name objects to the below object group** check box and select the group name from the drop-down box to import the objects directly to an existing distinguished name object group.

Step 7 Click **Import**.

Edit objects

Edit network objects and groups to update their settings or manage variable sets as required for your deployment.

Procedure

Step 1 Choose **Objects** > **Object Management**.

Step 2 Choose an object type from the list; see Objects, on page 2.

Step 3 Click **Edit** ( ) next to the object you want to edit.

If **View** ( ) appears instead, the object belongs to an ancestor domain and has been configured not to allow overrides, or you do not have permission to modify the object.

Step 4 Modify the object settings as desired.

Step 5 If you are editing a variable set, manage the variables in the set; see Manage variables, on page 121.

Step 6 For objects that can be configured to allow overrides:

- If you want to allow overrides for this object, check the **Allow Overrides** check box; see Allow object overrides, on page 14. You can change this setting only for objects that belong to the current domain.

- If you want to add override values to this object, expand the Override section and click **Add**; see Add object overrides, on page 14.

Step 7 Click **Save**.

Step 8 If you are editing a variable set, and that set is in use by an access control policy, click **Yes** to confirm that you want to save your changes.

What to do next

- If an active policy references your object, deploy configuration changes; see Deploy configuration changes.

View objects and their usage

This task enables you to view where objects are used within your system, helping you understand object dependencies and usage across policies and settings.

You can view usage details of objects on the Object Management page. Firewall Management Center provides this functionality for many object types. However, some object types are not supported.

Before you begin

No prerequisites are required to view object usage details.

Follow these steps to view objects and their usage:

Procedure

Step 1 Choose **Objects** > **Object Management**.

Step 2 Choose one of the following supported object types:

- Access List > Extended
- Access List > Standard
- AS Path
- Community List
- Interface
- Network
- Policy List
- Port
- Prefix List > IPv4 Prefix List
- Prefix List > IPv6 Prefix List
- Route Map

- SLA Monitor
- URL
- VLAN Tag

Step 3 Click the **Find Usage** ( ) icon next to the object.

The Object Usage window displays a list of all the policies, objects, and other settings where the object is in use. Click any of the listed items to know more about the object usage. For policies and some other settings where the object is used, you can click the corresponding links to visit the respective UI pages.

After completing these steps, you will be able to see where each object is used within your system, allowing you to manage and audit object usage efficiently.

Filter objects or object groups

Use this task when working with large configurations or when you need to identify and manage unused objects or object groups.

Procedure

Step 1 Choose **Objects** > **Object Management**.

Step 2 Enter your filter criteria in the **Filter** field.

The page updates as you type to display matching items.

You can use these wildcards:

- The asterisk (*) matches zero or more occurrences of a character.
- The caret (^) matches content at the beginning of a string.
- The dollar sign ($) matches content at the end of a string.

Step 3 To identify and review unused objects or object groups, check the **Show Unused Object** check box.

Note

- If an object is a part of an unused object group, the object is considered as used. However, the unused object group is displayed when the **Show Unused Object** check box is checked.
- The **Show Unused Object** check box is available only for network, port, URL, and VLAN tag object types.

Object groups

An object group is a configuration grouping that

- enables referencing multiple objects with a single configuration,
- allows objects and object groups to be used interchangeably in the web interface, and
- supports grouping of network, port, VLAN tag, URL, and PKI objects.

Object group usage and management

The system allows you to use objects and object groups interchangeably in the web interface. For example, anywhere you would use a port object, you can also use a port object group.

Network object groups can be nested. You can add a network object group to another network object group, up to 10 levels deep.

Objects and object groups of the same type cannot have the same name.

Edit an object group used in a policy, such as a network object group in an access control policy. Then, re-deploy the configuration for your changes to take effect.

Deleting a group does not delete the objects in the group; only their association with each other. Additionally, you cannot delete a group that is in use in an active policy. For example, you cannot delete a VLAN tag group that is used in a VLAN condition in a saved access control policy.

Object references and literal entries in object groups

Object groups can contain either object references or literal entries.

- Object references are links to existing Host/Network Objects. When the original object is modified (for example, an IP address change), all object groups containing references to that object are automatically updated to reflect the change.
- Literal entries are manually created entries within an object group. Unlike object references, literal entries are independent configuration entities and are not dynamically linked to existing Host/Network Objects. Therefore, changes made to a Host/Network Object do not propagate to object groups containing literal entries.

For example, an administrator may manually add a literal entry with the same name and IP address as an existing Host/Network Object. Although the entries appear identical, the literal entry remains independent of the original object. If the original Host/Network Object is modified, only object groups containing a reference to that object are updated.

Modifying a Host/Network Object updates only those object groups that contain a reference to that object. Object groups containing literal entries continue to display their original configuration unless the literal entries are manually modified.

Group reusable objects

Group reusable objects to simplify configuration and management in Secure Firewall Management Center.

- Organize object types such as Network, Port, URL, VLAN Tag, Distinguished Name, and PKI into groups for easier policy application.

This task is relevant when you need to manage multiple reusable objects and apply them efficiently in firewall policies.

Grouping objects allows for streamlined configuration and reduces manual effort when deploying or updating policies.

- Use this task when creating or editing object groups for policy deployment in Secure Firewall Management Center.

Before you begin

No explicit prerequisites are stated. Ensure you have access to Secure Firewall Management Center and the necessary permissions to manage objects.

- Identify the object types you want to group before starting the task.

Follow these steps to group reusable objects:

Procedure

Step 1 Choose **Objects** > **Object Management**.

Step 2 If the object type you want to group is **Network**, **Port**, **URL**, or **VLAN Tag**:

a) Choose the object type from the list of object types.
b) Choose **Add Group** from the **Add [Object Type]** drop-down list.

Step 3 If the object type you want to group is **Distinguished Name**:

a) Expand the **Distinguished Name** node.
b) Choose **Object Groups**.
c) Click **Add Distinguished Name Group**.

Step 4 If the object type you want to group is **PKI**:

a) Expand the **PKI** node.
b) Choose one of the following:
   - **Internal CA Groups**
   - **Trusted CA Groups**
   - **Internal Cert Groups**
   - **External Cert Groups**
c) Click **Add [Object Type] Group**.

Step 5 Enter a unique **Name**.

Step 6 Choose one or more objects from the list, and click **Add**.

You can also:

- Use the filter field **Search** ( ) to search for existing objects to include, which updates as you type to display matching items. Click **Reload** ( ) above the search field or click **Clear** ( ) in the search field to clear the search string.

- Click **Add** (+) to create objects on the fly if no existing objects meet your needs.

Step 7 Optionally for **Network**, **Port**, **URL**, and **VLAN Tag** groups:

- Enter a **Description**.
- Check the **Allow Overrides** check box to allow overrides for this object group; see Allow object overrides, on page 14.

Step 8 Click **Save**.

After completing these steps, the object group is created and available for use in firewall policies. You can reference the group in active policies and deploy configuration changes as needed.

What to do next

- If an active policy references your object group, deploy configuration changes; see Deploy configuration changes.

Object overrides

An object override is a configuration mechanism that

- enables the definition of alternate values for objects, which the system applies to specified devices,
- lets administrators create a smaller set of shared policies that can be used across devices without losing the flexibility to modify policies for individual devices, and
- allows modifications to an object for devices or domains that require different definitions.

Object override usage and supported object types

Administrators can create an object whose definition works for most devices and use overrides to specify modifications for those devices needing different definitions. You may also need to override an object for all devices. This approach allows you to create a single policy that applies everywhere. Object overrides allow you to use fewer shared policies and customize them for individual devices when necessary.

You can target overrides to a specific domain. The override value applies to all devices in that domain unless you set a device-specific override. The object manager enables selection of objects that can be overridden and allows listing device-level or domain-level overrides for each object.

Supported object types for overrides include:

- Network
- Port
- VLAN tag
- URL
- SLA Monitor

- Prefix List
- Route Map
- Access List
- AS Path
- Community List
- Policy List
- Cert Enrollment (PKI)
- Key Chain

In the object manager, the **Override** column flags objects that support overrides:

- Green checkmark: Overrides can be created, and none have been added yet.
- Red X: Overrides are not supported for the object.
- Number: Indicates the count of overrides added for the object (for example, "2" means two overrides have been defined).

Object override example

For example, to deny ICMP traffic to different departments, each connected to a different network, you can define an access control policy that includes a network object called Departmental Network. By enabling overrides for this object, it's possible to create policy variations on each relevant device that specify the actual network associated with that device.

Manage object overrides

Object overrides are used when you need to adjust the behavior of reusable objects for specific use cases, such as applying different values in different domains or policies. Use this task when you want to add, allow, delete, or edit object overrides in the Secure Firewall Management Center.

Procedure

Step 1 Choose **Objects** > **Object Management**.

Step 2 Choose from the list of object types; see Objects, on page 2.

Step 3 Click **Edit** ( ) next to the object you want to edit.

If **View** ( ) appears instead, the object belongs to an ancestor domain and has been configured not to allow overrides, or you do not have permission to modify the object.

Step 4 Manage the object overrides:

- Add—Add object overrides; see Add object overrides, on page 14.
- Allow—Allow object overrides; see Allow object overrides, on page 14.
- Edit—Edit object overrides; see Edit an object override, on page 15.
- Delete—In the object editor, click **Delete** ( ) next to the override you want to remove.

Step 5 Click **Save**.

Allow object overrides

Enable the Allow Overrides option for an object so that you can specify override values as required for different use cases.

Procedure

Step 1 In the object editor, check the **Allow Overrides** check box.

Step 2 Click **Save** to apply your changes.

What to do next

Add object override values; see Add object overrides, on page 14.

Add object overrides

Add or adjust object overrides so that network objects or groups can have different values assigned in specific domains or to specific devices, making configuration more flexible without modifying the base object.

Use this task when you need to assign different settings or values to the same object across multiple domains or devices—for example, when different parts of your organization require different configurations for the same network object.

Before you begin

Ensure that object overrides are allowed in your environment. See Allow object overrides, on page 14.

Procedure

Step 1 In the object editor, expand the **Override** section.

Step 2 Click **Add**.

Step 3 On **Targets**, choose domains or devices in the **Available Devices and Domains** list and click **Add**.

Step 4 On the Override tab, enter a **Name**.

Step 5 Optionally, enter a **Description**.

Step 6 Enter an override value.

Example:

```
For a network object, enter a network value.
```

Step 7 Click **Add**.

Step 8 Click **Save**.

What to do next

- If an active policy references your object, deploy configuration changes; see Deploy configuration changes.

Edit an object override

This task enables you to edit the description and value of an existing object override to update its configuration as needed.

You can modify the description and the value of an existing override, but you cannot modify the existing target list. Add a new override with new targets to replace the existing override.

Procedure

Step 1 In the object editor, expand the **Override** section.

Step 2 Click **Edit** (✎) next to the override you want to modify.

Step 3 Optionally, modify the **Description**.

Step 4 Modify the override value.

Step 5 Click **Save** to save the override.

Step 6 Click **Save** to save the object.

What to do next

- If an active policy references your object, deploy configuration changes; see Deploy configuration changes.

AAA servers

An AAA server is a network authentication, authorization, and accounting component that

- enables centralized management of user credentials
- supports policy enforcement, and
- provides accounting for user activities.

Add a RADIUS server group

RADIUS Server Group objects contain one or more references to RADIUS servers. These servers are used to authenticate users logging in through Remote Access VPN connections.

You can apply these objects to compatible Firewall Threat Defense devices.

Before you begin

Note You cannot override RADIUS Server Group Objects.

Procedure

Step 1 Select **Objects** > **Object Management** > **AAA Server** > **RADIUS Server Group**.

All currently configured RADIUS Server Group objects will be listed. Use the filter to narrow down the list.

Step 2 Choose and edit a listed RADIUS Server Group object, or add a new one.

See Configure RADIUS server options, on page 17 and RADIUS server group configuration options, on page 16 to configure this object.

Step 3 Click **Save**.

RADIUS server group configuration options

The configuration options for a RADIUS server group enable integration with identity policies and VPN authentication. This topic lists each option and describes its purpose, required settings, and usage notes.

Navigation path

Objects > **Object Management** > **AAA Server** > **RADIUS Server Group**. Choose and edit a configured RADIUS Server Group object or add a new one.

Fields

- **Name** and **Description**—Enter a name and optionally, a description to identify this RADIUS Server Group object.
- **Group Accounting Mode**—Defines how accounting messages are sent to the RADIUS servers. Choose one of these:
  - **Single**—Sends accounting messages to a single server (default).
  - **Multiple**—Sends accounting messages to all servers in the group simultaneously.
- **Retry Interval**—Sets the interval (1–10 seconds) between attempts to contact RADIUS servers.
- **Realms**(Optional)—Specify the Active Directory (AD) realm associated with this RADIUS server group. The selected realm is referenced in identity policies to determine VPN authentication identity sources

for relevant traffic flows. This realm effectively provides a bridge from the identity policy to this RADIUS server group. If no realm is associated with this RADIUS server group, the RADIUS server group cannot be reached to determine the VPN authentication identity source for a traffic flow in an identity policy.



Note This field is mandatory if you use remote access VPN with User Identity and RADIUS as the identity source.

- **Enable authorize only**—If the group is used only for authorization or accounting (not authentication), enable authorize-only mode. This mode omits the RADIUS server password from Access-Request messages; the configured password for individual RADIUS servers is ignored.
- **Enable interim account update** and **Interval**—Enables the generation of interim accounting update messages to notify the RADIUS server of newly assigned IP addresses. The Interval option sets the period (1 to 120 hours; default is 24 hours) between updates.
- **Enable Dynamic Authorization** and **Port**— Enables the RADIUS dynamic authorization or change of authorization (CoA) services for this RADIUS server group. Specify the listening port for RADIUS CoA requests in the **Port** field. The valid range is 1024 to 65535 and the default value is 1700. Once defined, the corresponding RADIUS server group will be registered for CoA notification and it listens to the port for the CoA policy updates from the Cisco Identity Services Engine (ISE).
- **Merge Downloadable ACL with Cisco AV Pair ACL**—Enables merging a downloadable access control list (dACL) with a Cisco attribute-value (AV) pair ACL.

  A downloadable ACL defines and updates access control lists in Cisco ISE. Administrators can then download the updated ACLs to all applicable controllers. For more information about using dACLs in Cisco ISE, see the chapter on Segmentation, section on authorization policies, in the *Cisco ISE Administrator Guide*.

  A Cisco AV pair ACL can be utilized to define specific authentication, authorization, and accounting elements for each individual session. For more information about using dACLs in Cisco ISE, see the chapter on Segmentation, section on authorization profile settings, in the *Cisco ISE Administrator Guide*.

  If you select **Merge Downloadable ACL with Cisco AV Pair ACL**, you can choose the following options:

  - **After Cisco AV Pair ACL**—Place downloadable ACL entries after AV pair entries.
  - **Before Cisco AV Pair ACL**—Place downloadable ACL entries before AV pair entries.

- **RADIUS Servers**—For server-specific configuration options, see Configure RADIUS server options, on page 17.

Related Topics

Add a RADIUS server group, on page 16

Configure RADIUS server options

Use this topic to configure a RADIUS server, including authentication, accounting, connectivity, and security settings.

Navigation path

Objects > **Object Management** > **AAA Server** > **RADIUS Server Group**. Choose and edit a listed RADIUS Server Group object, or add a new one. In the RADIUS Server Group dialog, choose and edit a RADIUS Server or add a new server.

Fields

- **IP Address/Hostname**—Specify the hostname or IP address (IPv4 or IPv6) of the RADIUS server that receives authentication requests. Only one server may be selected. To add servers, add more RADIUS Server entries to the RADIUS Server Group list.
- **RADIUS Server-Enabled Message Authenticator**—Message authenticators safeguard server-to-client communication and are required for a secure connection between your RADIUS server and firewall devices. Disabling message authenticators may expose your firewalls to potential attacks.

  Your RADIUS server must support and be configured for Message-Authenticator. Requires Firewall Threat Defense Version 7.2.10+, 7.4.3+, or 7.6.1+. For the Firepower 4100/9300, this feature may require an FXOS upgrade; for minimum builds, see *Cisco Secure Firewall Threat Defense Compatibility Guide*.
- **Authentication Port**—The port on which RADIUS authentication and authorization are performed. The default is 1812.
- **Key** and **Confirm Key**—Enter the shared secret to encrypt data between the managed device (client) and the RADIUS server.

  The key is a case-sensitive, alphanumeric string of up to 127 characters. Special characters are permitted.

  Enter the same key in this field and on the RADIUS server. Enter the shared secret again in the Confirm field.
- **Accounting Port**—Enter the port number for RADIUS accounting. The default port is 1813.
- **Timeout**—Specify the session timeout for authentication.

  Set the timeout value to 60 seconds or more for RADIUS two-factor authentication. The default timeout value is 10 seconds.
- **Connect Using** —Select the method to connect the device to a RADIUS server using either a route lookup or a specific interface.
  - Click the **Routing** radio button to use the routing table.
  - Click the **Specific Interface** radio button and choose a security zone/interface group or the Management interface (the default) from the drop-down list. If you want to use Management, you must choose it specifically; it is not available when using a route lookup. You cannot specify any other management-only interface as the RADIUS source.You can also choose a loopback interface group.
- **Redirect ACL**—Select the redirect ACL from the list or add a new one.

  Use the ACL name from the device to control redirected traffic. The Redirect ACL name here must be the same as the *redirect-ACL* name in ISE server. When you configure the ACL object, ensure that you select Block action for ISE and DNS servers, and Allow action for the rest of the servers.

Related Topics

Add a RADIUS server group, on page 16

RADIUS server group configuration options, on page 16

Add a single sign-on server

Add a single sign-on (SSO) server to integrate SAML-based authentication for remote access VPN users. This task enables secure user authentication through your identity provider.

Use this task when you need to configure SAML single sign-on for remote access VPN, allowing users to authenticate using an external identity provider. You typically need to perform this configuration when you integrate with SAML IdPs, such as Microsoft Azure.

Before you begin

Obtain the following from your SAML identity provider:

- Identity Provider Entity ID URL
- Sign-in URL
- Sign-out URL
- Identity provider certificate and enroll the certificate in Firewall Threat Defense using the Firewall Management Center web interface (**Devices** > **Certificates**)

For more information, see Configuring a SAML Single Sign-On Authentication.

Follow these steps to add a single sign-on server:

Procedure

Step 1 Choose **Objects** > **Object Management** > **AAA Server** > **Single Sign-on Server**.

Step 2 Click **Add Single Sign-on Server** and provide the following details:

Field	Description
Name	Enter the name for the SAML single sign-on server object.
Identity Provider Entity ID	Enter the URL that the SAML IdP defines to uniquely identify a service provider. Enter the URL for a metadata XML page that describes how the SAML Issuer responds to requests.
SSO URL	Enter the URL for signing in to the SAML identity provider server.
Logout URL	Enter the URL for signing out of the SAML identity provider server.
Base URL	Enter the URL that redirects users back to Firewall Threat Defense after the identity provider authenticates them. This should be the access interface URL configured for the Firewall Threat Defense remote access VPN.

Field	Description
Identity Provider Certificate	Enter the IdP certificate enrolled into the Firewall Threat Defense to verify the signed messages. Select an identity provider certificate from the list or click **Add** to create a new certificate enrollment object. For more information, see Manage Firewall Threat Defense certificates. You must enroll all of the Microsoft Azure registered application CA certificates as Trustpoints on the Firewall Threat Defense. The Microsoft Azure SAML identity provider is configured on Firewall Threat Defense for the initial application. All connection profiles are mapped to the configured MS Azure SAML identity provider. For each of the MS Azure applications (other than the default), you can choose the required trustpoint(CA certificate) in the connection profile configuration of the remote access VPN. For details, see Configure AAA Settings for Remote Access VPN.
Service Provider Certificate	Firewall Threat Defense certificate, which will be used to sign the requests and build circle of trust with IdP. If you have not enrolled internal Firewall Threat Defense certificates, click + to add and enroll a certificate. For more information, see Manage Firewall Threat Defense certificates.
Request Signature	Select the encryption algorithm to sign the SAML single sign-on requests. The signatures are listed from weakest to strongest: SHA1, SHA256, SHA384, SHA512. Select None to disable encryption.
Request Timeout	Specify how long users can complete the single sign-on request (SAML assertion validity duration). The SAML IdP has two time outs: *NotBefore* and *NotOnOrAfter*. The Firewall Threat Defense validates if its current time is within the time range of (lower limit) *NotBefore* and (upper limit) the smaller of *NotBefore* plus *timeout* and *NotOnOrAfter*. Thus, if you set a timeout longer than the IdP's *NotOnOrAfter* timeout, the specified timeout is ignored and the *NotOnOrAfter* timeout is selected. If the sum of the specified timeout and the *NotBefore* timeout is less than the *NotOnOrAfter* time, Firewall Threat Defense timeout overrides the timeout. The timeout range is 1-7200 seconds; the default is 300 seconds.
Enable IdP only accessible on Internal Network	Select this option if the SAML IdP resides on the internal network. Firewall Threat Defense acts as a gateway and establishes communication between the users and IdP using an anonymous webvpn session.
Request IdP re-authentication on Login	Select this option to authenticate user at each login even if the previous IdP session is valid.
Allow Overrides	Select this check box to allow overrides for this single sign-on server object.

Step 3 Click **Save**.

The single sign-on server is added and configured for SAML authentication. Users can now authenticate to the remote access VPN using the specified identity provider.

Access lists

An access list is a traffic selection object that

- identifies and selects traffic to which a service will apply
- is used when configuring features such as route maps for Firewall Threat Defense devices, and
- distinguishes between allowed and blocked traffic, where blocked traffic is excluded from the service but not necessarily dropped altogether.

Types of access lists

You can configure these types of ACL:

- Extended: Identifies traffic based on source and destination address and ports. Supports IPv4 and IPv6 addresses, which you can mix in a given rule.
- Standard: Identifies traffic based on destination address only. Supports IPv4 only.

Each access list is composed of one or more access control entries (ACEs), or rules. The order of ACEs is important.

1. When the ACL is evaluated, a packet is tested against each ACE in the order listed.
2. After a match is found, no more ACEs are checked.
3. Place more specific rules at the top of an ACL for correct behavior.

Place the most specific rules at the top of the ACL so it works as expected

Configure an extended ACL object

Configure extended ACL objects to define traffic classification rules based on source and destination addresses, protocols, ports, application groups, or IPv6 traffic. This enables granular control over network traffic for security and policy enforcement.

Use extended ACL objects when you want to match traffic based on source and destination addresses, protocol and port, application group, or if the traffic is IPv6. These objects are useful for policy-based routing and direct internet access scenarios, and provide flexibility in defining access control entries for various network requirements.

Procedure

Step 1 Choose **Objects** > **Object Management** > **Access List** > **Extended** from the table of contents.

Step 2 Do one of the following:

- Click **Add Extended Access List** to create a new object.

- Click **Edit** ( ) to edit an existing object.

Step 3 In the **New Extended Access List Object** dialog box, enter a name for the object (no spaces allowed) and configure the access control entries:

a) Do one of the following:

- Click **Add** to create a new entry.
- Click **Edit** ( ) to edit an existing entry.

b) Select the **Action**, whether to Allow (match) or Block (not match) the traffic criteria.



Note The **Logging**, **Log Level**, and **Log Interval** options are used for access rules only (ACLs attached to interfaces or applied globally). Because ACL objects are not used for access rules, leave these values at their defaults.

c) Configure the source and destination addresses on the **Network** tab using any of the following techniques:

- Select the desired network objects or groups from the Available list and click **Add to Source** or **Add to Destination**. You can create new objects by clicking the + button above the list. You can mix IPv4 and IPv6 addresses.
- Type an address in the edit box below the source or destination list and click **Add**. You can specify a single host address (such as 10.100.10.5 or 2001:DB8::0DB8:800:200C:417A), or a subnet (in 10.100.10.0/24 or 10.100.10.0 255.255.255.0 format, or for IPv6, 2001:DB8:0:CD30::/60).

d) Click the **Port** tab and configure the service using any of the following techniques.

- Select the desired port objects from the Available list and click **Add to Source** or **Add to Destination**. You can create new objects by clicking the + button above the list. The object can specify TCP/UDP ports, ICMP/ICMPv6 message types, or other protocols (including "any"). However, the source port, which you typically would leave empty, accepts TCP/UDP only. You cannot select port groups.

  For TCP/UDP, note that you must use the same protocol in both the source and destination fields, if you specify both. For example, you cannot specify a UDP source port and a TCP destination port.
- Type or select a port or protocol in the edit box below the source or destination list and click **Add**.



Note To get an entry that applies to all IP traffic, select a destination port object that specifies "all" protocols.

e) Click the **Application** tab and choose the applications that are to be grouped for the direct internet access policy.



Important

- You cannot configure applications for cluster devices. Hence, this tab is not applicable for cluster devices.
- Use extended ACL with applications only in policy-based routing. Do not use it in other policies as its behavior is unknown and not supported. Ensure migration of the realm/ISE configuration for policy-based routing that uses User Identity and SGT in extended ACL.

Note

- The **Available Applications** list displays a fixed set of pre-defined applications. This list is a subset of the applications that are available on the Access Control policy as only they can be detected by their first packet (FQDN end-points resolved to IP addresses and port). The application definitions are updated through the VDB updates and are pushed to Firewall Threat Defense during subsequent deployments.
- User-defined custom applications or group of applications are not supported.
- Currently, Firewall Management Center neither supports user-defined custom applications or group of applications nor allows you to modify the pre-defined applications list.
- You can use the filter options provided under the **Application Filters** to refine this list.

f) Click the **Users** tab and choose the users, user groups, or both that are to be classified for the Policy Based Routing (PBR).

Important Use extended ACL with users, user groups, or both only for Policy Based Routing. Do not use it in other policies as its behavior is unknown and not supported.

Note

- The **Available Realms** list displays the configured active directory/LDAP realms. For information on creation of realm and managing them, see Create an LDAP realm or an Active Directory realm and realm directory and Manage a realm respectively.

  Note Local realms and Azure AD realms are not supported.

- The **Available Users** list displays the downloaded users and user groups of the selected AD/LDAP realms. To download the users, user groups, or both, navigate to **Integrations** > **Other Integrations** > **Realms**, and then click **Download** against the relevant active directory/LDAP realms.

  Note Threat defense can support a maximum of 512 user groups and 64000 user-IP mappings.

- The user to IP mapping and user group membership information are updated and pushed to the Firewall Threat Defense from the Firewall Management Center during the user login or logouts, and changes in the group memberships.

g) Click the **Security Group Tag** tab and choose the source SGT tags to be classified for the direct internet access policy.

Important Use extended ACL with SGTs only for Policy Based Routing. Do not use it in other policies as its behavior is unknown and not supported.

Note

- The **Available Security Group Tags** list displays the configured security group tags. You can choose to use the ISE SGTs or create custom SGTs.
- To use ISE SGTs, ensure that ISE is integrated with Firewall Management Center with **Session Directory Topic** and SXP Topic subscribed. For more information on ISE integration, see Ways to configure the Cisco Identity Services Engine (Cisco ISE) identity source.

  Note The supported ISE versions are 3.2, 3,1, 3.0, and 2.7 patch 2+.

- For information on creating custom SGTs, see Create a security group tag object, on page 45.

h) Select the required application, and click **Add to Rule**.

Note

- Do not configure destination networks and applications in the extended ACL object.
- The selected applications (Network Service objects) in each of the access control entries, form a Network Service Group (NSG) and this group is deployed on the Firewall Threat Defense. The NSG is used in direct internet access to classify traffic based on the match with the selected application group.

i) Click **Add** to add the entry to the object.

Troubleshooting Tips

If necessary, click and drag the entry to move it up or down in the rule order to the desired location. Repeat the process to create or edit additional entries in the object.

Step 4 If you want to allow overrides for this object, check the **Allow Overrides** check box; see Allow object overrides, on page 14.

Step 5 Click **Save**.

Configure a standard ACL object

Use standard ACL objects when you only need to match traffic based on destination IPv4 address. Extended ACLs allow more granularity if needed.

Procedure

Step 1 Select **Objects** > **Object Management** > **Access List** > **Standard** from the table of contents.

Step 2 Do one of the following:

- Click **Add Standard Access List** to create a new object.
- Click **Edit** (✎) to edit an existing object.

Step 3 In the **New Standard Access List Object** dialog box, enter a name for the object (no spaces allowed), and configure the access control entries:

a) Do one of the following:

- Click **Add** to create a new entry.
- Click **Edit** (✎) to edit an existing entry.

b) For each access control entry, configure the following properties:

- **Action**—Whether to allow (match) or block (not match) the traffic criteria.
- **Network**—Add the IPv4 network objects or groups that identify the destination of the traffic.

c) Click **Add** to add the entry to the object.

d) If necessary, click and drag the entry to move it up or down in the rule order to the desired location.

Repeat the process to create or edit additional entries in the object.

Step 4 If you want to allow overrides for this object, check the **Allow Overrides** check box; see Allow object overrides, on page 14.

Step 5 Click **Save**.

Configure address pools

You can configure IP address pools for both IPv4 and IPv6 that can be used for clustering or for VPN remote access profiles. For clustering in Individual interface mode, you can also configure MAC address pools.

Procedure

Step 1 Select **Objects** > **Object Management** > **Address Pools**.

Step 2 Click **IPv4 Pools** and then **Add IPv4 Pools**, and configure the following fields.

- **Name**—Enter the name of the address pool. It can be up to 64 characters.
- **Description**—Add an optional description for this pool.

- **IP Address**—Enter a range of addresses available in the pool. Use dotted decimal notation and a dash between the beginning and the end address, for example: 10.10.147.100-10.10.147.177.
- **Mask**—Identifies the subnet on which this IP address pool resides.
- **Allow Overrides**—Check this check box to enable object overrides. Click the expand arrow to show the **Overrides** table. You can add a new override by clicking **Add**. See Object overrides, on page 12 for more information.

Step 3 Click **Save**.

Step 4 Click **IPv6 Pools** and then **Add IPv6 Pools**, and configure the following fields.

- **Name**—Enter the name of the address pool. It can be up to 64 characters
- **Description**—Add an optional description for this pool.
- **IPv6 Address**—Enter the first IP address available in the configured pool and the prefix length in bits. For example: 2001:DB8::1/64.
- **Number of Addresses**—Identifies the number of IPv6 addresses, starting at the Starting IP Address, that are in the pool.
- **Allow Overrides**—Check this check box to enable overrides. Click the expand arrow to show the **Overrides** table. You can add a new override by clicking **Add**. See Object overrides, on page 12 for more information.

Step 5 Click **Save**.

Step 6 Click **MAC Address Pool** and then **Add MAC Address Pool**, and configure the following fields.

For clustering in Individual interface mode, you can configure a MAC address pool for the interface. It is not common to manually configure MAC addresses for an interface, but if you have special needs to do so, then this pool is used to assign a unique MAC address to each interface. See Configure the MAC address.

- **Name**—Enter the name of the address pool. It can be up to 64 characters
- **Description**—Add an optional description for this pool.
- **MAC Address Range**—Enter a range of MAC addresses available in the pool. Use a dash between the beginning and the end address, for example: 000C.F142.4CD1-000C.F142.4CD7.
- **Allow Overrides**—Check this check box to enable overrides. Click the expand arrow to show the **Overrides** table. You can add a new override by clicking **Add**. See Object overrides, on page 12 for more information.

Create an application filter

An application filter is a reusable object that allows you to organize and manage applications efficiently for use in policies that require application matching.

An application filter does these:

- Simplifies application control and policy management.

- Groups applications based on type, risk, business relevance, category, and tags.
- Supports policies such as access control, QoS, intelligent application bypass, and decryption.

Application filter configuration options

- Use the system-provided filters to find applications. Select the desired applications and click **Add to Rule**.
- To remove an application from the filter, click the delete icon next to it.
- Click the info icon next to an application to view its information.

For detailed information on application characteristics, refer to Application rule conditions.

Configure an AS PATH object

Enable control over BGP routing updates and filtering by specifying a sequence of autonomous system (AS) numbers.

An AS Path is a mandatory attribute to set up BGP. It is a sequence of AS numbers through which a network can be accessed. AS path is a sequence of intermediate AS numbers between source and destination routers that form a directed route for packets to travel. Neighboring autonomous systems (ASes) use BGP to exchange and update messages about how to reach different AS prefixes. After each router makes a new local decision on the best route to a destination, it sends the route and associated path information, including distance metrics and path attributes, to its peers. As this information travels through the network, each router along the path prepends its unique AS number to a list of ASes in the BGP message. This list is the route's AS-PATH.

An AS path, along with an AS prefix, provides a specific handle for a one-way transit route through the network. Use the Configure AS Path page to create, copy, and edit autonomous system (AS) path policy objects. Create AS path objects to use when configuring route maps, policy maps, or BGP Neighbor Filtering. An AS path filter allows you to filter the routing update message by using regular expressions.

You can use this object with Firewall Threat Defense devices.

Procedure

Step 1 Choose **Objects** > **Object Management** > **AS Path** from the table of contents.

Step 2 Click **Add AS PATH**.

Step 3 Enter a name for the AS PATH object in the **Name** field. Valid values are between 1 and 500.

Step 4 Click **Add** on the **New AS PATH Object** window.

a) Select the Allow or Block options from the **Action** drop-down list to indicate redistribution access.
b) Specify the regular expression that defines the AS PATH filter in the **Regular Expression** field.
c) Click **Add**.

Step 5 If you want to allow overrides for this object, check the **Allow Overrides** check box; see Allow object overrides, on page 14.

Step 6 Click **Save**.

Configure a BFD template

Configure a BFD template to define BFD interval values and authentication settings for single-hop or multi-hop sessions. This enables flexible and reusable BFD configurations across interfaces.

The BFD template specifies a set of BFD interval values. BFD interval values as configured in the BFD template are not specific to a single interface. You can also configure authentication for single-hop and multi-hop sessions. Echo mode is disabled by default. You can enable Echo mode on single-hop only.

Before you begin

No prerequisites are required to configure a BFD template.

Procedure

Step 1 Choose **Objects** > **Object Management** > **BFD Template**.

Step 2 Click **Add BFD Template** or **Edit**.



Note If you are editing a template, you cannot modify its name and type.

Step 3 On the **Template** tab, configure the following:

- **Template Name**—The name of this BFD template. You must assign a name in order to configure the rest of the parameters in the template. The template name cannot have spaces and cannot have only numbers.
- **Type**—Click the **Single-Hop** or **Multi-Hop** radio button.
- **Enable Echo**—(Optional) Enables Echo for the single-hop template.

If the Echo function is not negotiated, BFD control packets are sent at a high rate to meet the detection time. If the Echo function is negotiated, BFD control packets are sent at a slower, negotiated rate and self-directed echo packets are sent at a high rate. We recommend that you use Echo mode, if possible.

Step 4 On the **Interval** tab, configure the following:

a) From the **Interval Type** drop-down list, select **Microseconds** or **Milliseconds**.
b) In the **Multiplier** field, enter the value to be used for computing the hold down time. This value indicates the number of consecutive BFD control packets that must be missed from a BFD peer before BFD declares that the peer is unavailable and the Layer 3 BFD peer is informed of the failure. The range is 3 to 50. The default is 3.
c) In the **Minimum Transmit** field, enter the minimum transmit interval capability. The range is 50 to 999 milliseconds or 50000 to 999000 microseconds.
d) In the **Minimum Receive** field, enter the minimum receive interval capability. The range is 50 to 999 milliseconds or 50000 to 999000 microseconds.

Step 5 On the **Authentication** tab, configure the following:

- **Authentication Type**—Select **NONE**, **md5**, **meticulous-sha-1**, **meticulous-md5**, or **sha-1** from the drop-down list.

- **Encrypted Password**—(Optional) Enables encryption of the authentication password.
- **Password**—The authentication password that must be sent and received in the packets using the routing protocol being authenticated. The valid value is a string containing 1 to 29 uppercase and lowercase alphanumeric characters, except that the first character cannot be a digit or a digit followed by a whitespace. For example, '1password' or '0 password' is invalid.
- **Key ID**—The shared key ID that matches the key value. The range is 0 to 255.

Step 6 Click **OK**.

Step 7 Click **Apply** to save the BFD template configuration.

The BFD template is configured with the specified interval values, authentication settings, and echo mode options. The template can now be applied to interfaces as needed.

Cipher suite lists

A cipher suite list is an object that

- contains several cipher suites,
- enables negotiation of SSL or TLS encrypted sessions, and
- can be used in SSL rules to control encrypted traffic based on the negotiated cipher suite.

Usage of cipher suite lists in SSL rules

You can use cipher suites and cipher suite lists in SSL rules to control encrypted traffic. These rules evaluate whether the client and server negotiated the SSL session using that cipher suite. If you add a cipher suite list to an SSL rule, and if an SSL session negotiates with any cipher suite from the list, the rule matches the session.

Note You can use cipher suites in the web interface wherever you use cipher suite lists. However, you cannot add, change, or remove cipher suites.

Create a cipher suite list

Create a custom cipher suite list to control which cipher suites are available for use in your environment.

Use this task when defining or updating the set of cipher suites to align with security policies or updated encryption standards.

Procedure

Step 1 Choose **Objects** > **Object Management** > **Cipher Suite List** from the list of object types.

Step 2 Click **Add Cipher Suites**.

Step 3 Enter a **Name**.

Step 4 Choose one or more cipher suites from the **Available Ciphers** list.

Step 5 Click **Add**.

Step 6 Optionally, click **Delete** ( ) next to any cipher suites in the **Selected Ciphers** list that you want to remove.

Step 7 Click **Save**.

What to do next

- If an active policy references your object, deploy configuration changes; see Deploy configuration changes.

Configure a community list

Configure a community list to group BGP destinations that share common attributes for route tagging and policy enforcement. This task enables you to create, copy, and edit community list policy objects for use in route maps or policy maps.

Community lists allow marking sets of prefixes for common routing policies, such as filtering or assigning local preference.

A community is an optional transitive BGP attribute used for route tagging. Communities are numerical values assigned to prefixes. They are advertised to neighbors. Upstream providers use these markers to apply routing policies, such as filtering or modifying attributes.

Use the Configure Community Lists page to create, copy, and edit community list policy objects. You can use this object with Firewall Threat Defense devices.

Community lists are used in match clauses of route maps and consist of ordered matching statements. Each destination is compared against the rules. The system stops processing further rules when it finds a match.

Procedure

Step 1 Choose **Objects** > **Object Management** > **Community List** from the table of contents.

Step 2 Click **Add Community List**.

Step 3 In the **Name** field, specify a name for the community list object.

Step 4 Click **Add** on the **New Community List Object** window.

Step 5 Select the **Standard** radio button to indicate the community rule type.

Standard community lists are used to specify well-known communities and community numbers.

Note You cannot have entries using Standard and entries using Expanded community rule types in the same Community List object.

a) Select the Allow or Block options from the **Action** drop-down list to indicate redistribution access.
b) In the **COMMUNITIES** field, specify a community number. Valid values can be from 1 to 4294967295 or from 0:1 to 65534:65535.
c) Select the appropriate **Route Type**.

- Internet—Select to specify the Internet well-known community. Routes with this community are advertised to all peers (internal and external).
- No Advertise—Select to specify the no-advertise well-known community. Routes with this community are not advertised to any peer (internal or external).
- No Export—Select to specify the no-export well-known community. Routes with this community are advertised to only peers in the same autonomous system or to only other sub-autonomous systems within a confederation. These routes are not advertised to external peers.

Step 6 Select the **Expanded** radio button to indicate the community rule type.

Expanded community lists filter communities with regular expressions, which specify patterns used to match community attributes.

a) Select the Allow or Block options from the **Action** drop-down list to indicate redistribution access.
b) Specify the regular expression in the **Expressions** field.

Step 7 Click **Add**.

Step 8 If you want to allow overrides for this object, check the **Allow Overrides** check box; see Allow object overrides, on page 14.

Step 9 Click **Save**.

Configure an extended community list

Create an extended community list to organize destinations by shared attributes, enabling precise BGP route filtering and policy object use.

An extended community is a larger group of destinations that share some common attribute. The BGP extended community list has attributes that can be used to mark a set of prefixes that share a common attribute. These markers are used in the match clause of a route map to filter the routes for implementing route leaks among virtual routers. You can also define policy list objects with the extended community list for filtering. Matching statements are organized in an ordered list within the extended community list. Routes are matched against the rules until a match is found with the specified route target (standard rule type) or regular expression (expanded rule type). Create and edit extended community list policy objects using the Extended Community page.



Note The extended community lists are applicable only for configuring import or export of routes.

You can use this object with Firewall Threat Defense devices.

Procedure

Step 1 Select **Objects** > **Object Management** > **Community List** > **Extended Community**, and then click **Add Extended Community List**.

Step 2 In the **Name** field, specify a name for the extended community list object. You can enter up to 80 characters for the name.

Step 3 Select the extended community rule type:

- To specify one or more route targets, click the **Standard** radio button .
- To specify regular expressions, click the **Expanded** radio button.

Note Use either Standard or Expanded extended community rule type in each Extended Community List object.

Step 4 Click **Add**.

Step 5 If you have selected **Standard** as the extended community rule type, specify the following:

a) In the **Sequence No** field, enter the order in which you want the rule to be executed.

The sequence number must be unique in the list.

b) From the **Action** drop-down list, if you want to permit routes that have matching route target that is specified here, select **Allow**; if you want to deny routes that have matching route target that is specified here, select **Block**.

c) In the **Route Target** field, specify a route target.

- You can add a single route target or a set of route targets separated by commas in a single entry. For example, *1:2,1:4,1:6*.
- Valid values can be from 1:1 to 65534:65535.
- You can have a maximum of 8 route targets in an entry.
- You cannot have redundant route target set across multiple entries. For example, say you want to configure *seq1* with *1:200,100:100,1:300* route targets, and *seq2* with *1:300,100:100,1:200* route targets. This results in redundant route target set and cannot be deployed.

Step 6 If you have selected **Expanded** as the extended community rule type, specify the following:

a) In the **Sequence No** field, enter the order in which you want the rule to be executed.

The sequence number must be unique in the list.

b) From the **Action** drop-down list, if you want to permit routes that have matching regular expression that is specified here, select **Allow**; if you want to deny routes that have matching regular expression that is specified here, select **Block**.

c) Specify the regular expression in the **Expressions** field.

- You can add a single route target or a set of route targets separated by a space in a single entry. For example, *^((16) | (18)):(.)$*.

- You can add a maximum of 16 regular expressions to an entry.
- You cannot have redundant regular expression set across multiple entries. For example, say you want to configure *seq1* with *^((16) | (18)):(.)$ ^4_[0-9]*$* route targets, and *seq2* with *^4_[0-9]*$ ^((16) | (18)):(.)$* route targets. This results in redundant regular expression set and cannot be deployed.

For details on BGP regular expressions, see here.

Step 7 If you want to allow overrides for this object, check the **Allow Overrides** check box; see Allow object overrides, on page 14.

Step 8 Click **Save**.

The extended community list can be referenced in the match clause of the Route Map object or Policy List object:

- In the Route Map object, the name of the extended community list is displayed in the **Add Route Map Entry** > **Match Clause** > **BGP** > **Community List** > **Add Extended Community List** dialog. For more details on configuring BGP settings in a route map, see Configure a route map, on page 84.
- In the Policy List object, the name of the extended community list is displayed in the **Add Policy List** > **Community Rule** > **Add Extended Community List** dialog. For more details on configuring BGP settings in a policy list, see Configure a policy list, on page 78.

DHCP IPv6 pools

A DHCP IPv6 pool is a configuration element that

- defines a range of IPv6 addresses for DHCP assignment,
- is referenced when configuring DHCPv6 services, and
- enables address allocation to IPv6 clients.

For clients that use StateLess Address Auto Configuration (SLAAC) in conjunction with the Prefix Delegation feature (Enable the IPv6 prefix delegation client), you can configure the Firewall Threat Defense to provide information such as the DNS server or domain name when they send Information Request (IR) packets to the Firewall Threat Defense by defining a DHCP IPv6 Pool and assigning it to the DHCPv6 server. The Firewall Threat Defense only accepts IR packets and does not assign addresses to the clients. You will configure the client to generate its own IPv6 address by enabling IPv6 autoconfiguration on the client. Enabling stateless autoconfiguration on a client configures IPv6 addresses based on prefixes received in Router Advertisement messages; in other words, based on the prefix that the Firewall Threat Defense received using Prefix Delegation.

To add a pool, see Create a DHCP IPv6 pool.

Distinguished names

A distinguished name is a certificate attribute that

- represents the subject or issuer for a public key certificate,

- can be used in rules to control encrypted traffic based on certificate negotiation, and
- consists of attributes such as country code, common name, organization, and organizational unit.

A *distinguished name group* is a named collection of existing distinguished name objects.

Distinguished name attributes and usage

Distinguished name objects and distinguished name groups can be used in TLS/SSL rules to control encrypted traffic based on whether the client and server negotiated the TLS/SSL session using a server certificate with the distinguished name as subject or issuer.

A distinguished name can include a country code, common name, organization, and organizational unit. In most cases, the certificate includes only a common name. For example, the common name in the certificate for `https://www.cisco.com` is `cisco.com`. The certificate can include multiple Subject Alternative Names (SANs). You can use these names as distinguished names (DNs) in a rule condition. For detailed information about SANs, see RFC 5280, section 4.2.1.6.

The format of a distinguished name object that references a common name is `CN=`*`name`*. If you add a DN rule condition without `CN=`, the system prepends `CN=` before saving the object.

The system uses Server Name Indication (SNI) to match the DN in the TLS/SSL rule whenever possible.

You can also add a distinguished name with one of each of the attributes listed in this table, separated by commas.

Table 1: Distinguished name attributes

Attribute	Description	Allowed Values
C	Country Code	two alphabetic characters
CN	Common Name	up to 64 alphanumeric, backslash (/), hyphen (-), quotation ("), or asterisk (*) characters, or spaces
O	Organization	up to 64 alphanumeric, backslash (/), hyphen (-), quotation ("), or asterisk (*) characters, or spaces
OU	Organizational Unit	up to 64 alphanumeric, backslash (/), hyphen (-), quotation ("), or asterisk (*) characters, or spaces

DN rule conditions

- When the system first detects an encrypted session to a new server, DN data is not available for ClientHello processing. As a result, the first session might not be decrypted.

  If the server requests TLS 1.3, the setting for TLS server identity discovery can help by making sure the server certificate is known before making decryption policy decisions. For more information, see TLS server identity discovery.

- You *cannot* configure a distinguished name condition if you also choose the **Decrypt - Known Key** action. Because that action requires you to choose a server certificate to decrypt traffic, the certificate already matches the traffic.

Wildcard examples

You can define one or more asterisks (*) as wildcards in an attribute. In a common name attribute, you can define one or more asterisks per domain name label. Wildcards match only in that label, but you can define multiple labels with wildcards. See this table for examples.

Table 2: Common name attribute wildcard examples

Attribute	Matches	Does not match
CN=*ample.com	example.com	mail.example.com example.text.com ampleexam.com
CN=exam*.com	example.com	mail.example.com example.text.com ampleexam.com
CN=*xamp*.com	example.com	mail.example.com example.text.com ampleexam.com
CN=*.example.com	mail.example.com	www.myhost.example.com example.com example.text.com ampleexam.com

Note The DN object `CN=amp.cisco.com` would *not* match a CN like `CN=auth.amp.cisco.com`, which is why wildcards are recommended in these cases.

For more information and examples, see Conditions for distinguished name rules in decryption policies.

Related Topics

Conditions for distinguished name rules in decryption policies

Create a distinguished name object

Enables you to create distinguished name objects, which uniquely identify and reference network entities within Secure Firewall Management Center.

Distinguished name objects are used to specify unique identifiers for certificates or access policies. Perform this task when configuring objects that require a distinguished or common name for identification.

Procedure

Step 1 Choose **Objects** > **Object Management**.

Step 2 Expand the **Distinguished Name** node, and choose **Individual Objects**.

Step 3 Click **Add Distinguished Name**.

Step 4 Enter a **Name**.

Step 5 In the **DN** field, enter a value for the distinguished name or common name. You have the following options:

- If you add a distinguished name, you can include one of each attribute listed in Distinguished names, on page 33 separated by commas.
- If you add a common name, you can include multiple labels and wild cards.

Step 6 Click **Save**.

What to do next

- If an active policy references your object, deploy configuration changes; see Deploy configuration changes.

DNS server groups

A DNS server group is a network configuration object that

- contains multiple Domain Name System (DNS) servers
- resolves fully qualified domain names (FQDNs), such as `www.example.com`, to IP addresses, and
- enables efficient domain name resolution within a network.

Create a DNS server group object

Create and manage a group of DNS servers for use in network policies to simplify DNS server assignments.

DNS server group objects allow you to define multiple DNS servers that can be referenced by other configuration elements. Use this task to centralize DNS server assignments for interfaces or policies.

Procedure

Step 1 Choose **Objects** > **Object Management** > **DNS Server Group** from the network objects list.

Step 2 Click **Add DNS Server Group**.

Step 3 Enter a **Name**.

Step 4 Optionally, enter the **Default Domain** that will be used to append to the host names that are not fully-qualified.

This setting is only used for the default server group.

Step 5 The default **Timeout** and **Retries** values are pre-populated. Change these values if necessary.

- Retries—Specify how many times (0 to 10) to retry contacting DNS servers if there is no response. Default is 2.
- Timeout—Specify how many seconds (1 to 30) to wait before trying the next DNS server. Default is 2 seconds; this timeout doubles with each retry.

Step 6 Enter the **DNS Servers** IP addresses (IPv4 or IPv6), separated by commas. *(Maximum of 6 servers per group.)*

Step 7 Click **Save**.

What to do next

Assign the configured DNS server group to interface objects in the DNS platform settings. For more information, see Configure DNS server settings.

External attributes

Dynamic objects

A dynamic object is a network object that

- specifies one or many IP addresses retrieved using REST API calls or the Dynamic Attributes Connector from cloud sources,
- can be used in access control rules without deploying dynamic changes, and
- automatically updates object values on managed devices after being pushed by the Dynamic Attributes Connector.

Kinds of dynamic objects

There are these kinds of dynamic objects:

- Connector-created dynamic objects: The system pushes dynamic objects created using the dynamic attributes connector to the Firewall Management Center as soon as you create them. The system updates them at regular intervals.
- API-created dynamic objects have the following characteristics:
  - Are IP addresses, with or without classless inter-domain routing (CIDR), that can be used in access control rules much like a network object.
  - Do not support fully-qualified domain names or address ranges.
  - You must update these objects using an API.

For more information about API-created dynamic objects, see API-created dynamic objects, on page 44.

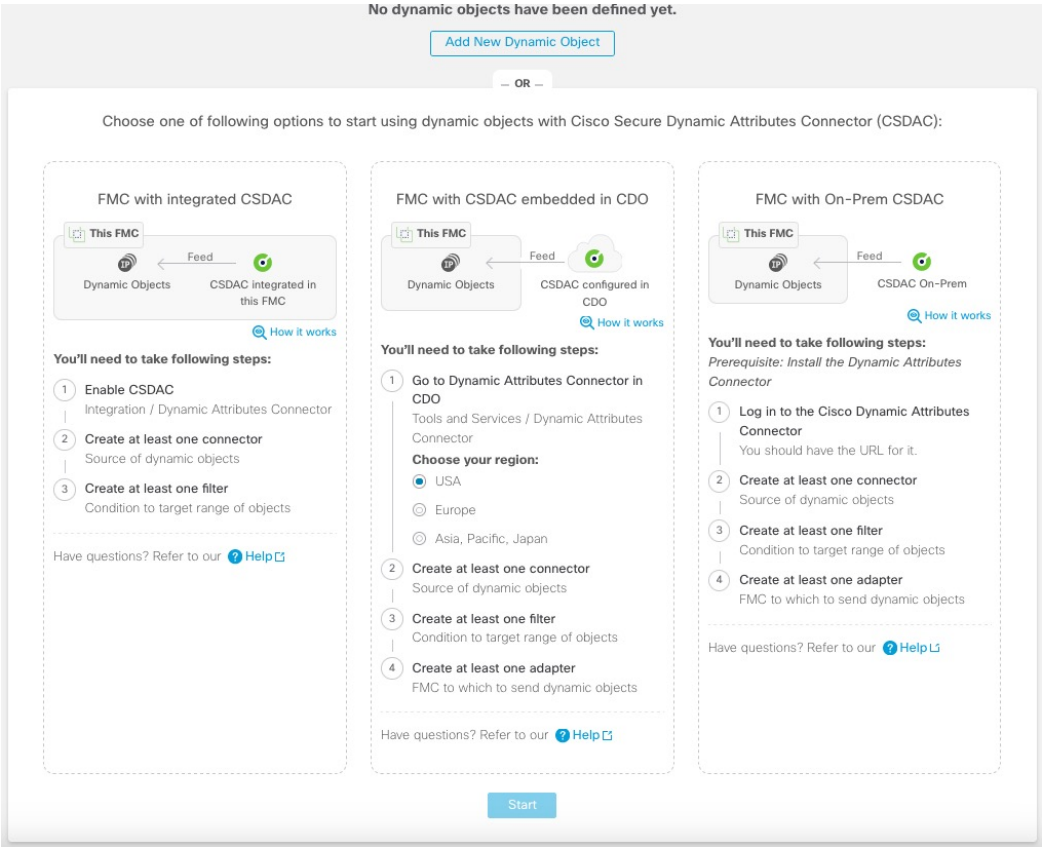
Note Unlike most other objects, dynamic objects do not have to be deployed to managed devices to take effect. Add a dynamic object to the **Dynamic Attributes** tab in a rule and deploy the rule. The object values are automatically updated on the managed device as soon as possible after being pushed by the Dynamic Attributes Connector.

Create dynamic objects for the first time

This topic explains how to create dynamic objects for the first time, describes the informational page displayed when no dynamic objects exist, and outlines the steps to begin the configuration process using the secure firewall manager or dynamic attributes connector.

The page available at **Objects** > **Object Management** > **External Attributes** > **Dynamic Object** is displayed as follows if you have not configured any dynamic objects yet.

If you have already created some dynamic objects, see Dynamic objects page features, on page 43.



To use this page:

- Select the section that matches how you plan to deploy the dynamic attributes connector.

  Deployment method and outcome:

Method	Outcome
Cloud-Delivered Firewall Management Center with CSDAC Embedded	Access dashboard and configure within application
Cloud-Delivered Firewall Management Center with or CSDAC	Download connector and follow guided configuration
Management Center with Integrated CSDAC	Enable and configure connector within on-prem manager

- Click **How it works** to view diagrams and get more help with that type of deployment.

  The following topics provide more information.

  - Create dynamic objects with embedded Dynamic Attributes Connector, on page 39
  - Dynamic objects with Security Cloud Control, on page 40
  - Create dynamic objects with on-premises Dynamic Attributes Connector, on page 41

- Click **Start** to begin the configuration process in the related application:

  - If you clicked the **FMC with Integrated CSDAC** method, **Start** takes you to **Objects** > **Object Management** > **External Attributes** > **Dynamic Object** where you can enable and configure the Dynamic Attributes Connector.

    For more information, see Enable the dynamic attributes connector.

  - If you clicked the **FMC with CSDAC embedded in** Security Cloud Control method, **Start** starts Security Cloud Control.

    For more information, see the guide titled *Managing Firewall Threat Defense with Cloud-delivered Firewall Management Center in Security Cloud Control*.

  - If you clicked the **FMC with On-Prem CSDAC** method, **Start** enables you to download the Dynamic Attributes Connector if you have not done so already.

    For more information, see the *Cisco Secure Dynamic Attributes Connector Configuration Guide*.

Create dynamic objects with embedded Dynamic Attributes Connector

A dynamic object is a firewall configuration element that

- enables integration with the embedded Dynamic Attributes Connector
- allows retrieval of IP addresses from cloud services, and
- supports use in access control and DNS rules within the Secure Firewall Management Center.

Integration and configuration details

The Dynamic Attributes Connector is already integrated with the Secure Firewall Management Center (**Integration** > **Dynamic Attributes Connector**).

To use the dynamic attributes connector included with the secure firewall manager, enable it (Integration > Dynamic Attributes Connector), configure connectors that receive dynamic objects, and set up dynamic attributes filters that determine what dynamic objects are sent to this secure firewall manager.

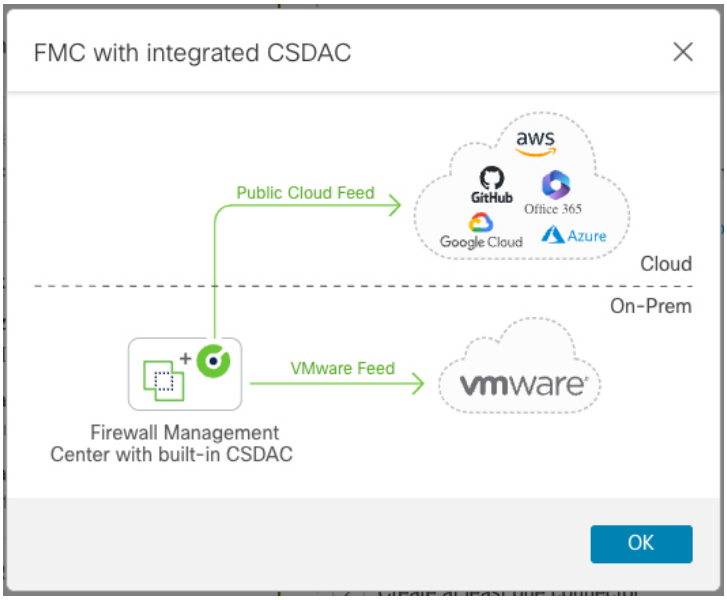
To use this type of deployment:

1. Enable the Dynamic Attributes Connector as discussed in Enable the dynamic attributes connector.
2. Configure connectors, which retrieve IP addresses from cloud services.

   For more information, see Connectors.
3. Configure dynamic attributes filters, which determine what IP addresses to send to the management center.

   For more information, see Create dynamic attributes filters.
4. View your dynamic objects at **Objects** > **Object Management** > **External Attributes** > **Dynamic Object**.
5. Use dynamic objects in access control rules (**Policies** > **Access Control heading** > **Access Control**, then click the **Dynamic Attributes** tab).

Figure 1: Dynamic attributes connector integration

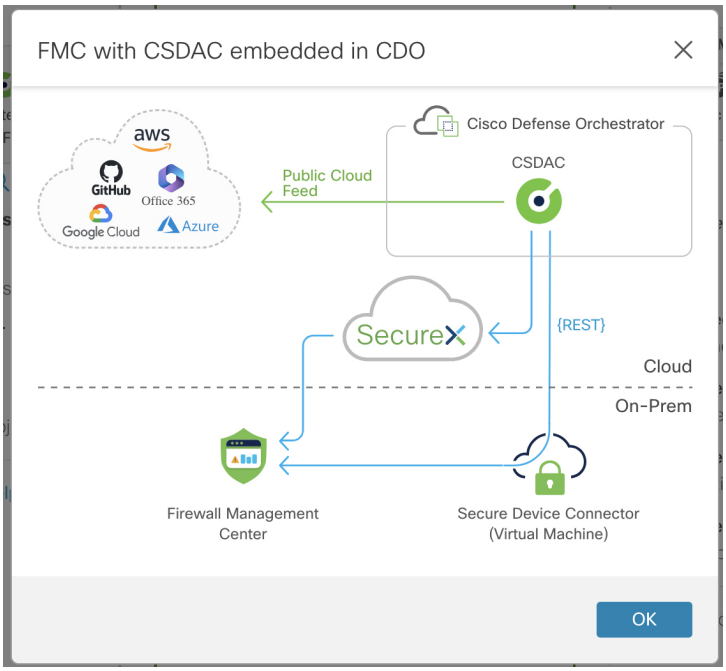


Dynamic objects with Security Cloud Control

A dynamic object is a network management feature that

- enables the retrieval of IP addresses from cloud services using connectors,
- allows you to create dynamic attributes filters to determine which objects are sent, and
- supports integration with secure firewall management centers through adapters.

This page is displayed if you configure the Dynamic Attributes Connector provided with Security Cloud Control.



The diagram includes details about configuring Security Cloud Control that are not discussed in this guide. For more information, see Secure Device Connector (SDC) or SecureX and Security Cloud Control.

To use this type of deployment:

1. Configure connectors to retrieve IP addresses from cloud services. For instructions, see Connectors.
2. Define dynamic attribute filters to specify which IP addresses are sent to the management center.

   For more information, see Create dynamic attributes filters.
3. Set up adapters to send IP addresses to a Secure Firewall Management Center or Cloud-Delivered Firewall Management Center.

   For more information, see the section on creating adapters in the *Cisco Security Cloud Control: Secure Firewall Threat Defense Management*.
4. Log in to the Secure Firewall Management Center that you defined as an adapter.

   If the Secure Firewall Management Center is managed by Security Cloud Control, access the **Cloud-Delivered FMC**.
5. View your dynamic objects at **Objects** > **Object Management** > **External Attributes** > **Dynamic Object**.
6. Use dynamic objects in access control rules (**Policies** > **Access Control heading** > **Access Control**, then click the **Dynamic Attributes** tab).

Create dynamic objects with on-premises Dynamic Attributes Connector

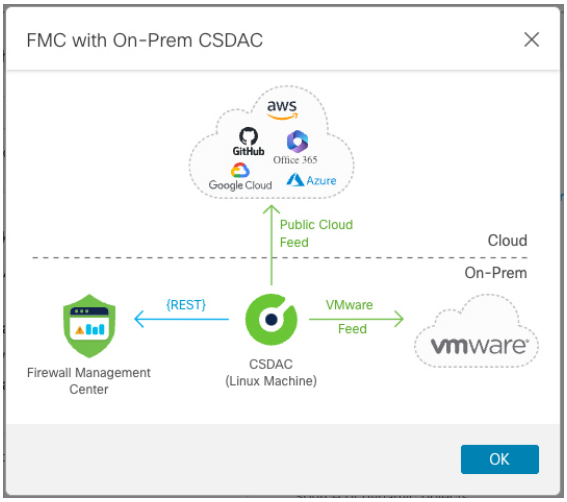
A dynamic object deployment is a configuration method that

- installs the Dynamic Attributes Connector on a supported Linux virtual machine,

- configures connectors, adapters, and dynamic attributes filters to retrieve and send IP addresses to a management center, and
- enables use of dynamic objects in access control and DNS rules.

This page appears if you indicated that you are configuring the on-premises Dynamic Attributes Connector to send dynamic objects to a Secure Firewall Management Center or Cloud-Delivered Firewall Management Center.

Figure 2: Dynamic attributes connector configuration



The on-premises Dynamic Attributes Connector can send dynamic objects to a Secure Firewall Management Center or Cloud-Delivered Firewall Management Center.

To use this type of deployment, complete these steps:

1. Install the Dynamic Attributes Connector on a supported Linux virtual machine.
2. Configure connectors, which retrieve IP addresses from cloud services.

   For more information, see the section on creating connectors in the *Cisco Secure Dynamic Attributes Connector Configuration Guide*.
3. Configure adapters, which send IP addresses to a Secure Firewall Management Center or Cloud-Delivered Firewall Management Center.

   For more information, see the section on creating adapters in the *Cisco Secure Dynamic Attributes Connector Configuration Guide*.
4. Configure dynamic attributes filters, which determine what IP addresses to send to the management center.

   For more information, see the section on configuring dynamic attributes filters in the *Cisco Secure Dynamic Attributes Connector Configuration Guide*.
5. You can view your dynamic objects at **Objects** > **Object Management** > **External Attributes** > **Dynamic Object**.
6. Use dynamic objects in access control rules (**Policies** > **Access Control heading** > **Access Control**, then click the **Dynamic Attributes** tab).

For more information, see the *Cisco Secure Dynamic Attributes Connector Configuration Guide*.

Dynamic objects page features

The Dynamic Objects page displays all dynamic objects created using the API or the Cisco Secure Dynamic Attributes Connector. You can edit, delete, or view IP address mappings on this page. The page enables you to view or download IP addresses associated with each object.

The page available at **Objects** > **Object Management** > **External Attributes** > **Dynamic Object** is displayed similar to this if you have already configured some dynamic objects.



This page displays information about each dynamic object and enables you to view or download IP addresses associated with that object. For more information, see Dynamic object mappings, on page 43.

Dynamic object mappings

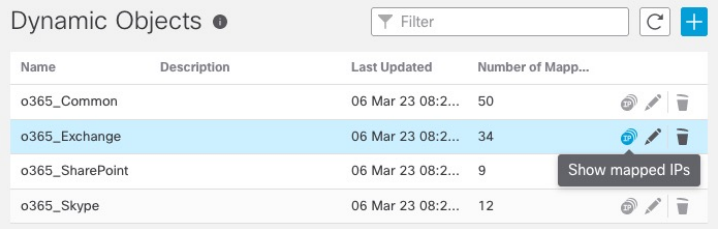
A dynamic object mapping is a configuration feature that

- enables connectors to send IPs matching dynamic attribute filters to the Firewall Management Center,
- allows users to view or download a current list of mapped IP addresses, and
- updates IP addresses dynamically over time to reflect changes in access control rules.

Mapped IP address information

If you configured dynamic objects through the API or by using the dynamic attributes connector, the connectors send IPs that match dynamic attribute filters to the Firewall Management Center at regular intervals.

To view or download a current list of these IP addresses, click **Show Mapped IDs** in the Firewall Management Center, as illustrated.



IP addresses are added dynamically over time. Review the mapped IP addresses regularly, especially if the access control rules do not behave as expected.

Related topics

- API-created dynamic objects, on page 44
- About the dynamic attributes connector

API-created dynamic objects

An API-created dynamic object is a network object that

- specifies one or more IP addresses retrieved using REST API calls or the Dynamic Attributes Connector,
- can be used in access control rules without deploying dynamic changes to the objects, and
- updates IP addresses from cloud sources.

For more information about the dynamic attributes connector, see the *Cisco Secure Dynamic Attributes Configuration Guide* (link to guide).

Differences between dynamic objects and network objects

- Dynamic objects created using the dynamic attributes connector are pushed to the Firewall Management Center as soon as they are created. These objects are then updated at regular intervals.
- API-created dynamic objects:
  - are IP addresses, with or without classless inter-domain routing (CIDR), that can be used in access control rules much like a network object,
  - do not support fully-qualified domain names or address ranges, and
  - must be updated using an API.

Add or edit an API-created dynamic object

This task enables you to add or edit a dynamic object using the API, allowing you to group IP addresses for use in access control rules. Dynamic objects help automate address management and streamline policy configuration.

A dynamic object is a group of IP addresses, with or without classless inter-domain routing (CIDR). You can use dynamic objects in access control rules much like a network object. This procedure is relevant when you need to manage dynamic address groups via the API instead of manually creating static objects.

Note This procedure is not necessary if you use the Dynamic Attributes Connector because it automatically creates dynamic objects for you.

Before you begin

Consult the *Firepower Management Center REST API Quick Start Guide* for information about how to use the object services REST API to populate the IP object with an address. Dynamic objects do not require deployment.

Follow these steps to add or edit an API-created dynamic object:

Procedure

Step 1 Click **Objects** > **Object Management**.

Step 2 Click **External Attributes** > **Dynamic Objects**.

Step 3 Click **Add Dynamic Object** or **Edit** ( ).

Step 4 Enter a **Name** for the object and an optional **Description**.

Step 5 From the **Type** list, click **IP**.

What to do next

If necessary, update the dynamic object using the API. Deployment is not required.

Security group tags

A security group tag is a network security object that

- specifies a single SGT value,
- is used in rules to control traffic with SGT attributes not assigned by Cisco ISE, and
- cannot be grouped or overridden.

Related Topics

Autotransition from custom SGTs to ISE SGTs
Custom SGT conditions
ISE SGTs versus custom SGT rule conditions

Create a security group tag object

This task allows you to add a security group tag object to enforce network access policies based on SGT.

You can create security group tag objects only in the global domain. A Control license is required for Classic devices. Any Smart License provides coverage for Smart Licensed devices.

Before you begin

- Disable ISE or ISE-PIC connections because custom SGT objects cannot be created if either is used as an identity source.

  Click **Integration** > **Other Integrations** > **Identity Sources**, click **None**, then click **Save**.

- See Best practices for ISE/ISE-PIC integration for guidelines for using the identity source.

Follow these steps to create a security group tag object:

Procedure

Step 1 Click **Objects** > **Object Management**.

Step 2 Click **External Attributes** > **Security Group Tag**.

Step 3 Click **Add Security Group Tag**.

Step 4 Enter a **Name**.

Step 5 Optionally, enter a **Description**.

Step 6 In the **Tag** field, enter a single SGT.

Step 7 Click **Save**.

What to do next

- If an active policy references your object, deploy configuration changes; see Deploy configuration changes.

File lists

A file list is a threat management feature that

- allows you to manually specify files using unique SHA-256 hash values,
- overrides AMP cloud verdicts to improve detection accuracy within your environment, and
- supports categorization into clean or custom detection lists, enabling precise control over file treatment.

If you use malware defense and the AMP cloud incorrectly identifies a file's disposition, you can add the file to a file list to improve future detection. Specify these files using SHA-256 hash values. You can add up to 10,000 unique SHA-256 values in each file list.

File list categories

File lists are organized into two predefined categories.

- **Clean list**: If you add a file to this list, the system treats it as if the AMP cloud assigned a clean disposition.
- **Custom detection list**: If you add a file to this list, the system treats it as if the AMP cloud assigned a malware disposition.

Because you manually specify the blocking behavior for the files included in these lists, the system does not query the AMP cloud for these files' dispositions. You must configure a rule in the file policy with either the **Malware Cloud Lookup** or **Block Malware** action and a matching file type to calculate a file's SHA value.

Caution Do **not** include malware on the clean list. The clean list overrides both the AMP cloud disposition and the custom detection list.

Source files for file lists

Source files allow you to add multiple SHA-256 values to a file list by uploading a comma-separated value (CSV) file. The system validates the contents and populates the file list with valid SHA-256 values. Upload source files must meet the following requirements:

- The source file must be a simple text file with a .CSV extension.
- Any header must start with a pound sign; it is treated as a comment and will not be uploaded.

- Each entry should contain a single SHA-256 value followed by a description, and must end with either LF or CR+LF newline character.
- The system ignores any additional information present in the entry.

Note the following behaviors and limitations:

- Deleting a source file from the file list also removes all associated SHA-256 hashes from the list.
- You cannot upload multiple files to a file list if the upload would result in more than 10,000 distinct SHA-256 values in the list.
- The system truncates descriptions exceeding 256 characters to the first 256 characters when uploading. If the description contains commas, you must use an escape character (`\,`). If no description is provided, the source file name is used instead.
- All non-duplicate SHA-256 values are added to the file list. If a SHA-256 value already exists, it is not modified by subsequent uploads. Any threat name or description is derived from the individual SHA-256 value in related events.
- Invalid SHA-256 values in a source file are not uploaded.
- If multiple uploaded source files contain entries for the same SHA-256 value, the system uses the most recent value.
- If a source file contains multiple entries for the same SHA-256 value, the system uses the last entry.
- You cannot edit a source file directly within the object manager. To make changes, edit your local source file, delete the existing file from the system, then upload the revised file.
- The number of entries associated with a source file refers to the number of distinct SHA-256 values. Deleting a source file decreases the file list's SHA-256 entries by the number of valid entries in the deleted file.

Add individual SHA-256 values to file lists

Add unique SHA-256 values to file lists to enhance file detection and management within the system.

You must have the Malware Defense license for this procedure.

Adding SHA-256 values identifies files for detection and prevents duplicates by ensuring only unique values are included.

Before you begin

- Right-click a file or malware event from the event view, choose **Show Full Text** in the context menu, and copy the full SHA-256 value for pasting into the file list.

Follow these steps to add individual SHA-256 values to file lists:

Procedure

Step 1 Choose **Objects** > **Object Management**.

Step 2 Choose **File List** from the list of object types.

Step 3 Click **Edit** ( ) next to the clean list or custom detection list where you want to add a file.

If **View** ( ) appears instead, the object belongs to an ancestor domain, or you do not have permission to modify the object.

Step 4 Choose `Enter SHA Value` from the **Add by** drop-down list.

Step 5 Enter a description of the source file in the **Description** field.

Step 6 Enter or paste the file's entire value in the **SHA-256** field. The system does not support matching partial values.

Step 7 Click **Add**.

Step 8 Click **Save**.

What to do next

- If an active policy references your object, deploy configuration changes; see Deploy configuration changes.

Note After configuration changes are deployed, the system no longer queries the AMP cloud for files on the list.

Upload an individual file to a file list

You must have the Malware Defense license for this procedure.

If you have a copy of the file you want to add to a file list, you can upload the file to the Secure Firewall Management Center for analysis. The system automatically calculates the file's SHA-256 value on upload. No file size limit applies for SHA-256 calculation.

Procedure

Step 1 Choose **Objects** > **Object Management**.

Step 2 Choose **File List** from the list of object types.

Step 3 Click **Edit** ( ) next to the clean list or custom detection list where you want to add a file.

If **View** ( ) appears instead, the object belongs to an ancestor domain, or you do not have permission to modify the object.

Step 4 From the **Add by** drop-down list, choose **Calculate SHA**.

Step 5 (Optional) Enter a description of the file in the **Description** field. If you leave this field blank, the file name becomes the description during upload.

Step 6 Click **Browse**, and choose a file to upload.

Step 7 Click **Calculate and Add SHA** to upload the file and calculate the SHA-256 value.

Step 8 Click **Save**.

After completing these steps, the file is uploaded, its SHA-256 value is calculated, and the file is added to the selected file list for analysis.

What to do next

- If an active policy references your object, deploy configuration changes; see Deploy configuration changes.

Note After you deploy configuration changes, the system no longer queries the AMP cloud for files on the list.

Upload source files to file lists

This task enables you to upload source files to file lists, allowing for efficient management and use of file data within the system.

You must have the Malware Defense license for this procedure.

Use this task when you need to add values from a source file to an existing file list in the system.

Procedure

Step 1 Choose **Objects** > **Object Management**.

Step 2 Click **File List**.

Step 3 Click **Edit** ( ) next to the file list where you want to add values from a source file.

If **View** ( ) appears instead, the object belongs to an ancestor domain, or you do not have permission to modify the object.

Step 4 In the **Add by** drop-down list, choose `List of SHAs`.

Step 5 (Optional) Enter a description of the source file in the **Description** field. If you do not enter a description, the system uses the file name.

Step 6 Click **Browse** to browse to the source file, then click **Upload and Add List**.

Step 7 Click **Save**.

After you complete these steps, the system uploads the source file and adds its values to the selected file list.

What to do next

- If an active policy references your object, deploy configuration changes; see Deploy configuration changes.

Note After you deploy the policies, the system stops querying the AMP cloud for files on the list.

Edit SHA-256 values in file lists

You must have the Malware Defense license for this procedure.

You can edit or delete individual SHA-256 values on a file list. However, you cannot directly edit a source file within the object manager. To make changes, first modify your source file, delete the copy on the system, and then upload the modified file.

Procedure

Step 1 Choose **Objects** > **Object Management**.

Step 2 Click **File List**.

Step 3 Click **Edit** ( ) next to the clean list or custom detection list where you want to modify a file.

If **View** ( ) appears instead, the object belongs to an ancestor domain, or you do not have permission to modify the object.

Step 4 You can:

- Click **Edit** ( ) next to the SHA-256 value you want to change, and modify the **SHA-256** or **Description** values as desired.
- Click **Delete** ( ) next to the SHA-256 value you want to delete.

Step 5 Click **Save** to update the file entry in the list.

Step 6 Click **Save** to save the file list.

What to do next

- If an active policy references your object, deploy configuration changes; see Deploy configuration changes.

Note After configuration changes are deployed, the system no longer queries the AMP cloud for files on the list.

Download source files from a file list

This task allows you to download source files from file lists, enabling you to obtain files for analysis or operational needs.

You must have the Malware Defense license for this procedure.

Procedure

Step 1 Choose **Objects** > **Object Management**.

Step 2 Choose **File List** from the list of object types.

Step 3 Click **Edit** ( ) next to the clean list or custom detection list where you want to download a source file.

If **View** ( ) appears instead, the object belongs to an ancestor domain, or you do not have permission to modify the object.

Step 4 Next to the source file you want to download, click **View** ( ).

Step 5 Click **Download SHA List** and follow the prompts to save the source file.

Step 6 Click **Close**.

FlexConfig objects

A FlexConfig object is a configuration element that

- provides customized configuration of features on Firewall Threat Defense devices
- enables configuration of features not otherwise available through Secure Firewall Management Center, and
- uses policy objects in FlexConfig policies to customize device behavior.

FlexConfig object types

You can configure these types of objects for FlexConfig.

- **Text objects** define free-form text strings used as variables in a FlexConfig object. These objects can have a single value or multiple values. Several predefined text objects are included in FlexConfig objects. To customize device behavior, edit the contents of the text object when using a FlexConfig object. When editing a predefined object, create device overrides for each device instead of changing the default values. This helps prevent configuration issues for other users who use the same FlexConfig object. For information on configuring text objects, see Configure FlexConfig text objects.
- **FlexConfig objects** include device configuration commands, variables, and scripting language instructions. During deployment, the system processes these instructions to create configuration commands with customized parameters for the target devices. You can add instructions at the beginning (prepended) of the configuration or at the end (appended), based on your requirements. Append any FlexConfig that depends on Secure Firewall Management Center-configured objects (for example, a network object) to the configuration deployment. This ensures the necessary objects are configured before the FlexConfig refers to them. For more information on configuring FlexConfig objects, see Configure FlexConfig objects.

Geolocation objects

A geolocation object is a network policy element that

- represents one or more countries or continents identified as the source or destination of traffic on your monitored network,

- can be used in various places in the system's web interface, including access control policies, SSL policies, remote access VPN, and event searches, and
- enables you to write access control rules that block traffic to or from certain countries.

Geolocation database updates

To ensure that you are using up-to-date information to filter your network traffic, we strongly recommend that you regularly update your Geolocation Database (GeoDB).

Create a geolocation object

Create a geolocation object to define groups of countries and continents for use in security policies.

Use geolocation objects in access control and policy rules to manage network traffic based on geographic criteria.

Procedure

Step 1 Choose **Objects** > **Object Management**.

Step 2 Choose **Geolocation** from the list of object types.

Step 3 Click **Add Geolocation**.

Step 4 Enter a **Name**.

Step 5 Select the countries and/or continents to include.

- Selecting a continent includes all countries within that continent, including future additions via GeoDB updates.
- You can select any combination of countries and continents.

Step 6 Click **Save**.

What to do next

- If an active policy references your object, deploy configuration changes; see Deploy configuration changes.

Interfaces

An interface is a network connection point that

- can be assigned to a *security zone*, an *interface group*, or both,
- enables application of security policies based on zones or groups, and
- supports flexible traffic control between different network segments.

Security zone and interface group assignments

Each interface can be assigned to a *security zone*, an *interface group*, or both. Apply your security policy based on zones or groups. For example, assign the "inside" interface to the "inside" zone and the "outside" interface to the "outside" zone. Configure the access control policy to allow traffic from inside to outside and block traffic from outside to inside. Some policies support only security zones. Other policies support both zones and groups.

For more information about interface objects, see Security zones and interface groups.

To add interface objects, see Create security zone and interface group objects.

Key chains

Key chains help you rotate authentication keys for IGP peers to enhance data security, prevent unauthorized users from guessing routing protocol authentication keys, and maintain uninterrupted, secure communication by allowing overlapping key lifetimes.

Key chain authentication and key management

Use rotating keys in key chains, with a duration of 180 days or less, to protect your devices and data. Rotating keys apply only to the OSPFv2 protocol.

Note Only the MD5 cryptographic algorithm is used for authentication.

Each key in a key chain has two lifetimes:

- Accept lifetime: The time interval within which the device accepts the key during key exchange with another device.
- Send lifetime: The time interval within which the device sends the key during key exchange with another device.

During a key's send lifetime, the device sends routing update packets using that key. If the key is not within the accept lifetime, the device does not accept communication from other devices.

If lifetimes are not configured, the system treats the configuration as using an MD5 authentication key without defined timelines.

When configuring authentication for routing protocols that provide key chains, configure the keys in your key chain with overlapping lifetimes to prevent losing secure communication if no active key is available.

Key selection in a key chain is determined as follows:

- When a key chain has more than one valid key, OSPF selects the key with the longest lifetime.
- The system prefers a key with an infinite lifetime.
- If keys have the same lifetime, then key with the higher key ID is preferred.

Create a key chain object

Create key chain objects that securely manage and organize authentication keys for devices on your network.

Use this task when you need to define and manage key chains for secure authentication. Key chains enable secure key exchange, helping prevent unauthorized access and ensuring reliable communication between devices.

Procedure

Step 1 Choose **Objects** > **Object Management**.

Step 2 Choose **Key Chain** from the list of object types, and then click **Add Key Chain**.

Step 3 In the Add Key Chain Object dialog box, do the following:

a. Enter a name for the key chain in the **Name** field.

The name must start with an underscore or letter, followed by alphanumeric characters or one of the following special characters: hyphen, underscore, plus sign, or period.

b. Add a key to the key chain, click **Add**.

c. Specify the key identifier (between 0 and 255; use 0 only to signal an invalid key) in the **Key ID** field.

d. The **Algorithm** field and the **Crypto Encryption Type** field displays the supported algorithm and the encryption type, namely MD5 and Plain Text.

e. Enter the password in the **Crypto Key String** field and confirm it in the appropriate field. Passwords can be up to 80 characters and must not be a single digit or start with a digit followed by a space (e.g., "0 pass" or "1" are invalid).

f. Set the **Accept Lifetime** and **Send Lifetime** values to define the time intervals for key exchange. The times default to UTC. When specifying start and end lifetimes, ensure that the start is earlier than the end and is never left blank if the end is entered.

g. Click **Add** to create keys. Create at least two keys with overlapping lifetimes to prevent loss of secure communication.

Repeat steps 3b to 3g to create keys.

Step 4 Manage overrides for the object:

- If you want to allow overrides for this object, check the **Allow Overrides** check box; see Allow object overrides, on page 14.
- If you want to add override values to this object, expand the Override section and click **Add**; see Add object overrides, on page 14.

Step 5 Click **Save**.

After you save, you can use the key chain object in device authentication and secure communication configurations.

What to do next

- If an active policy references your object, deploy configuration changes; see Deploy configuration changes.

Networks

A network object is a network entity that

- represents one or more IP addresses,
- can be used in access control policies, network variables, identity rules, network discovery rules, event searches, and reports, and
- is filtered automatically to show only valid objects for each configuration option.

Network object types

You can define a network object as one of these types:

- **Host**: A single IP address.

  IPv4 example:

  `209.165.200.225`

  IPv6 example:

  `2001:DB8::0DB8:800:200C:417A` or `2001:DB8:0:0:0DB8:800:200C:417A`

- **Range**: A range of IP addresses.

  IPv4 example:

  `209.165.200.225-209.165.200.250`

  IPv6 example:

  `2001:db8:0:cd30::1-2001:db8:0:cd30::1000`

- **Network**: An address block, also known as a subnet.

  IPv4 example:

  `209.165.200.224/27`

  IPv6 example:

  `2001:DB8:0:CD30::/60`

Note Security Intelligence ignores IP address blocks using a `/0` netmask.

- **FQDN**: A single fully-qualified domain name (FQDN). You can limit FQDN resolution to IPv4 address only, IPv6 address only, or both types. FQDNs must begin and end with a digit or letter. They can contain only letters, digits, and hyphens internally.

Example:

```
www.example.com
```

Note You can use FQDN objects in access control rules and prefilter rules, or manual NAT rules, only. The rules match the IP address obtained for the FQDN through a DNS lookup. To use an FQDN network object, ensure you have configured the DNS server settings in DNS server groups, on page 36 and the DNS platform settings in Configure DNS server settings.

You *cannot* use FDQN network objects in identity rules.

- **Group**: A group of network objects or other network object groups. You can create nested groups by adding a network object group to another. Nesting is supported up to 10 levels.

Note You can add up to 100 network literals in a network object. Each nested network object group contains up to 100 network literals.

Network wildcard masks

A network wildcard mask is an IP address mask that

- uses a discontinuous mask of bits to define network objects,
- enables you to create network objects with expanded subnet IP addresses, and
- appears as **Network Wildcard** in the **Type** column of the network object list.

Network wildcard mask usage and support

You can create and manage wildcard mask objects from the Network Wildcard Mask page in **Objects** > **Object Management** .

Standard network objects use contiguous masks, while wildcard network objects use discontinuous masks.

This table shows examples of IP addresses and indicates whether they are considered network wildcard objects:

Table 3: Network wildcard mask examples

Example IP address	Network wildcard?	Object type
192.0.0.0/8	No	Network
10.10.0.0/255.255.0.0	No	Network
10.10.0.10/255.255.0.255	Yes	Network Wildcard
72.0.240.10/255.255.240.255	Yes	Network Wildcard

Note Network wildcard objects and groups containing network wildcard objects are allowed only when configuring these policies:

- Prefilter policy
- Access control policy
- NAT policy

Guidelines and limitations

- To create network wildcard objects, in the Firewall Management Center UI, choose **Objects** > **Object Management** > **Network** and click **Add Network** and then **Add Object**. Select the **Network** option and enter the value as expanded subnet mask fotmat, such as 10.0.10.10 or 255.255.0.255.
- Features supported: Object override, group object support, group object override, wildcard literals, and wildcard object import.
- Network wildcard objects are supported only for IPv4 addresses.
- Network wildcard objects are supported from Firewall Management Center and Firewall Threat Defense versions 7.1 onwards.
- Network wildcard objects are supported only for Snort-3.

Create a network object

Create network objects to define network addresses, address ranges, or groups that can be used in security policies and access rules.

Network objects allow you to efficiently manage and reuse network definitions across your security policies. This task is relevant when you need to add, clone, or modify network objects in your management center.

Procedure

Step 1 Choose **Objects** > **Object Management**, and then Choose **Network** from the list of object types.

Step 2 Choose **Add Object** from the **Add Network** drop-down menu.

Step 3 Alternatively, you can clone an existing network object and edit the parameters to create a new network object. Click the **Clone** icon on the existing network object that you want to clone.

Step 4 Enter a **Name** and optionally, enter a **Description**.

Step 5 In the **Network** field, select the required option and enter an appropriate value; see Networks, on page 55.

Note You can add up to 100 network literals in a network object. Additionally, each nested network object group can contain a maximum of 100 network literals.

Step 6 (FQDN objects only) Select the DNS resolution from the **Lookup** drop-down menu to determine whether you want the IPv4, IPv6, or both IPv4 and IPv6 addresses associated with the FQDN.

Step 7 Manage overrides for the object:

- If you want to allow overrides for this object, check the **Allow Overrides** check box; see Allow object overrides, on page 14.
- If you want to add override values to this object, expand the Override section and click **Add**; see Add object overrides, on page 14.

Step 8 Click **Save**.

What to do next

- If an active policy references your object, deploy configuration changes; see Deploy configuration changes.

Import network objects

For details on importing network objects, see Import objects, on page 4.

PKI objects

PKI objects are public key certificates and paired private keys that

- represent the security components required to support your deployment
- consist of certificate authority certificates for internal and trusted CA objects
- consist of server certificates for internal and external certificate objects, and
- contain private keys paired with certificates for internal objects.

PKI object usage for applications and SSL rules

PKI objects support various security configurations across your network deployment.

You can use PKI objects for these applications:

- ISE/ISE-PIC identity source: Use trusted certificate authority objects and internal certificate objects to configure a connection to ISE/ISE-PIC as an identity source.
- Captive portal authentication: Use internal certificate objects to authenticate the identity of your captive portal device when connecting to users' web browsers.
- Secure LDAP or AD connections: Use trusted certificate authority objects to configure realms for secure connections to LDAP or AD servers.

If you use PKI objects in SSL rules, you can match traffic encrypted with:

- the certificate in an external certificate object
- a certificate either signed by the CA in a trusted CA object, or within the CA's chain of trust

If you use PKI objects in SSL rules, you can decrypt:

- outgoing traffic by re-signing the server certificate with an internal CA object
- incoming traffic using the known private key in an internal certificate object

You can manually input certificate and key information, upload a file containing that information, or in some cases, generate a new CA certificate and private key.

When you view a list of PKI objects in the object manager, the system displays the certificate's Subject distinguished name as the object value. Hover your pointer over the value to view the full certificate Subject distinguished name. To view other certificate details, edit the PKI object.



Note The Firewall Management Center and managed devices encrypt all private keys stored in internal CA objects and internal certificate objects with a randomly generated key before saving them. If you upload private keys that are password protected, the appliance decrypts the key using the user-supplied password, then reencrypts it with the randomly generated key before saving it.

PKI objects for certificate enrollment

A certificate enrollment object is a PKI configuration item that

- contains the Certification Authority (CA) server information and enrollment parameters required for creating Certificate Signing Requests (CSRs)
- obtains identity certificates from the specified CA, and
- may include certificate revocation information.

Internal certificate authority objects

An internal certificate authority object is a security certificate object that

- represents the CA public key certificate of a CA your organization controls
- consists of the object name, CA certificate, and paired private key, and
- can be used in SSL rules to decrypt outgoing encrypted traffic by re-signing the server certificate with the internal CA.

Internal certificate authority object usage and management

You can create an internal certificate authority object in these ways:

- import an existing RSA-based or elliptic curve-based CA certificate and private key
- generate a new self-signed RSA-based CA certificate and private key
- generate an unsigned RSA-based CA certificate and private key. You must submit a certificate signing request (CSR) to another CA to sign the certificate before using the internal CA object.

Note If you reference an internal CA object in a **Decrypt - Resign** SSL rule and the rule matches an encrypted session, the user's browser may warn that the certificate is not trusted while negotiating the SSL handshake. To avoid this, add the internal CA object certificate to either the client or domain list of trusted root certificates.

After you create an internal CA object containing a signed certificate, you can download the CA certificate and private key. The system encrypts downloaded certificates and private keys with a user-provided password.

Whether system-generated or user-created, you can modify the internal CA object name, but cannot modify other object properties.

You cannot delete an internal CA object that is in use. Additionally, after you edit an internal CA object used in an SSL policy, the associated access control policy goes out-of-date. You must re-deploy the access control policy for your changes to take effect.

CA certificates and private key import

A CA certificate and private key import is a security configuration process that

- enables uploading of X.509 v3 CA certificates and private keys,
- supports files encoded in DER or PEM formats, and
- validates that the certificate and key are properly paired before saving the object.

Supported file formats and import details

You can configure an internal CA object by importing an X.509 v3 CA certificate and private key. You can upload files encoded in one of these supported formats:

- Distinguished Encoding Rules (DER)
- Privacy-enhanced Electronic Mail (PEM)

If the private key file is password-protected, you can supply the decryption password. If the certificate and key are encoded in the PEM format, you can also copy and paste the information.

The system only allows you to upload files that contain proper certificate or key information. It also ensures the certificate and key are paired with each other. The system validates the pair before saving the object.

Note If you configure a rule with the **Decrypt - Resign** action, the rule matches traffic based on the referenced internal CA certificate's encryption algorithm type, in addition to any configured rule conditions. You must upload an elliptic curve-based CA certificate to decrypt outgoing traffic encrypted with an elliptic curve-based algorithm, for example.

Import a CA certificate and private key

Use this task when you need to add a new internal CA to your Secure Firewall Management Center for certificate management.

Administrators typically perform this task during initial setup or when updating trusted certificate authorities. It is relevant for those managing PKI infrastructure within Secure Firewall Management Center.

Procedure

Step 1 Choose **Objects** > **Object Management**.

Step 2 Expand the **PKI** node, and choose **Internal CAs**.

Step 3 Click **Import CA**.

Step 4 Enter a **Name**.

Step 5 Above the **Certificate Data** field, click **Browse** to upload a DER or PEM-encoded X.509 v3 CA certificate file.

Step 6 Above the **Key** field, click **Browse** to upload a DER or PEM-encoded paired private key file.

Step 7 If the uploaded file is password-protected, check the **Encrypted, and the password is:** check box, and enter the password.

Step 8 Click **Save**.

What to do next

- If an active policy references your object, deploy configuration changes; see Deploy configuration changes.

Generate a new CA certificate and private key

Generate a self-signed RSA-based CA certificate and accompanying private key by configuring an internal CA object.

Use this task to establish a trusted CA for your environment. The CA certificate enables secure signing and validation of other certificates and will be valid for ten years, with the **Valid From** date starting one week before creation.

Procedure

Step 1 Choose **Objects** > **Object Management**.

Step 2 Expand the **PKI** node, and choose **Internal CAs**.

Step 3 Click **Generate CA**.

Step 4 Enter a **Name**.

Step 5 Enter your identification attributes.

Step 6 Click **Generate self-signed CA**.

Signed certificates

A signed certificate is a digital security credential that

- is obtained from a certificate authority (CA)
- enables configuration of an internal CA object, and
- allows referencing the internal CA object in an SSL rule if it contains a signed certificate.

Certificate configuration steps

You can configure an internal CA object by obtaining a signed certificate from a CA. This process consists of these actions:

- Provide identification information to configure the internal CA object. This action generates an unsigned certificate, produces a paired private key, and creates a certificate signing request (CSR) to a specified CA.
- Upload the signed certificate issued by the CA to the internal CA object. The internal CA object now contains a signed certificate instead of the unsigned certificate.

Only reference an internal CA object in an SSL rule if it contains a signed certificate.

Create an unsigned CA certificate and CSR

Create an unsigned CA certificate and generate a certificate signing request (CSR) to enable secure certificate management in Secure Firewall Management Center.

Perform this task when you need to generate a new unsigned CA certificate and CSR, typically for setting up internal certificate authorities or renewing certificates for secure communications. Always complete this task before uploading a signed certificate issued by a certificate authority.

Procedure

Step 1 Choose **Objects** > **Object Management**.

Step 2 Expand the **PKI** node, and choose **Internal CAs**.

Step 3 Click **Generate CA**.

Step 4 Enter a **Name**.

Step 5 Enter the identification attributes.

Step 6 Click **Generate CSR**.

Step 7 Copy the CSR to submit to a CA.

Step 8 Click **OK**.

What to do next

- Upload a signed certificate issued by a CA as described in Upload a signed certificate issued in response to a CSR, on page 63

Upload a signed certificate issued in response to a CSR

Perform this task after receiving a signed certificate from a certificate authority (CA) in response to a CSR generated by the system. Once uploaded, the certificate is available for referencing in SSL rules.

Before you begin

Obtain the signed certificate file (DER or PEM-encoded X.509 v3 CA certificate) that was issued in response to the CSR.

Procedure

Step 1 Choose **Objects** > **Object Management**.

Step 2 Expand the **PKI** node, and choose **Internal CAs**.

Step 3 Click **Edit** ( ) next to the CA object containing the unsigned certificate awaiting the CSR.

Step 4 Click **Install Certificate**.

Step 5 Click **Browse** to upload a DER or PEM-encoded X.509 v3 CA certificate file.

Step 6 If the uploaded file is password protected, check the **Encrypted, and the password is:** check box, and enter the password.

Step 7 Click **Save** to upload a signed certificate to the CA object.

What to do next

- If an active policy references your object, deploy configuration changes; see Deploy configuration changes.

CA certificates and private key downloads

A CA certificate and private key download is a security operation that

- enables you to back up or transfer a CA certificate and its paired private key by downloading a file from an internal CA object,
- requires the system to decrypt the private key before creating the downloadable file, and
- uses a password provided by the user to encrypt the downloaded file for secure storage.

Encryption and storage of downloaded CA certificates and private keys

When you download a CA certificate and its private key, the system decrypts the private key before generating the file. Provide a password to encrypt the downloaded file. This helps keep it safe during storage and transfer. The system also encrypts the private key in the CA object with a randomly generated key before saving it to disk.

Caution Store downloaded key information in a secure location.

Caution If you download private keys as part of a system backup, they are decrypted and stored in the unencrypted backup file.

Download a CA certificate and private key

Obtain a CA certificate and private key required for certificate backup, migration, or integration within your secure firewall environment.

You can download CA certificates for both the current domain and ancestor domains. Use this task when you need to obtain the CA certificate and private key for backup, migration, or integration purposes.

Procedure

Step 1 Choose **Objects** > **Object Management**.

Step 2 Expand the **PKI** node, and choose **Internal CAs**.

Step 3 Select the internal CA object whose certificate and private key you want to download. Click **Edit** ( ).

Step 4 Click **Download**.

Step 5 Enter an encryption password in the **Password** and **Confirm Password** fields.

Step 6 Click **OK**.

Trusted certificate authority objects

A trusted certificate authority object is a security configuration item that

- represents a CA public key certificate belonging to a trusted CA,
- consists of the object name and CA public key certificate, and
- can be used in SSL policy, realm configurations, and integration with ISE or ISE-PIC.

Usage scenarios for trusted certificate authority objects

You can use trusted certificate authority objects and groups in the following configurations:

- SSL policy: Control traffic encrypted with a certificate signed either by the trusted CA, or any CA within the chain of trust.
- Realm configurations: Establish secure connections to LDAP or Active Directory (AD) servers.
- ISE or ISE-PIC connection: Select trusted certificate authority objects for the pxGrid Server CA and MNT Server CA fields.

After you create the trusted CA object, you can modify the name and add certificate revocation lists (CRL), but cannot modify other object properties. There is no limit on the number of CRLs you can add to an object. If you want to modify a CRL you have uploaded to an object, you must delete the object and recreate it.

Note Adding a CRL to an object has no effect when the object is used in your ISE or ISE-PIC integration configuration.

You cannot delete a trusted CA object that is in use. Additionally, after you edit a trusted CA object that is in use, the associated access control policy goes out-of-date. You must re-deploy the access control policy for your changes to take effect.

Trusted CA objects

A trusted CA object is a security object that:

- enables configuration of an external CA object by uploading an X.509 v3 CA certificate
- supports uploading files encoded in DER or PEM formats, and
- requires the system to validate the certificate before saving the object.

You can upload a file encoded in one of these supported formats:

- Distinguished Encoding Rules (DER)
- Privacy-enhanced Electronic Mail (PEM)

If the file is password-protected, you must supply the decryption password. If the certificate is encoded in the PEM format, you can also copy and paste the certificate information.

You can upload a CA certificate only if the file contains proper certificate information; the system validates the certificate before saving the object.

Add a trusted CA object

Add a trusted CA object so the system can validate certificates issued by a Certificate Authority (CA).

Use this task when you need to import a new trusted CA certificate into the system, such as when enabling secure communications or integrating with external certificate authorities.

Before you begin

- Have the DER or PEM-encoded X.509 v3 CA certificate file available.
- If the file is password-protected, ensure you have the password ready.

Procedure

Step 1 Choose **Objects** > **Object Management**.

Step 2 Expand the **PKI** node, and choose **Trusted CAs**.

Step 3 Click **Add Trusted CAs**.

Step 4 Enter a **Name**.

Step 5 Click **Browse** to upload a DER or PEM-encoded X.509 v3 CA certificate file.

Step 6 If the file is password-protected, check the **Encrypted, and the password is:** check box, and enter the password.

Step 7 Click **Save**.

The trusted CA object is added and available for use in certificate validation processes.

What to do next

- If an active policy references your object, deploy configuration changes; see Deploy configuration changes.

Certificate revocation lists in trusted CA objects

A certificate revocation list is a security file that

- can be uploaded to a trusted CA object
- enables control of encrypted traffic based on whether the issuing CA has revoked a certificate, and
- supports files encoded in Distinguished Encoding Rules (DER) or Privacy-enhanced Electronic Mail (PEM) formats.

Certificate revocation list usage in trusted CA objects

You can upload CRLs to a trusted CA object. If you reference that trusted CA object in an SSL policy, you can control encrypted traffic based on whether the CA that issued the session encryption certificate subsequently revoked the certificate.

Supported file formats for CRLs include:

- Distinguished Encoding Rules (DER)
- Privacy-enhanced Electronic Mail (PEM)

After you add the CRL, you can view the list of revoked certificates. If you want to modify a CRL you have uploaded to an object, you must delete the object and recreate it.

You can upload only files that contain a proper CRL. There is no limit to the number of CRLs you can add to a trusted CA object. However, you must save the object each time you upload a CRL, before adding another CRL.



Note Adding a CRL to an object has no effect when the object is used in your ISE/ISE-PIC integration configuration.

Add a certificate revocation list to a trusted CA object

Add a certificate revocation list (CRL) to a trusted certificate authority (CA) object. This step ensures that revoked certificates are recognized during certificate validation.

Add a CRL to a trusted CA object to ensure that Secure Firewall Management Center validates client certificates against a list of revoked certificates.

Note Adding a CRL to an object has no effect when the object is used in your ISE/ISE-PIC integration configuration.

Before you begin

Ensure you have a DER or PEM-encoded CRL file available for upload.

Procedure

Step 1 Choose **Objects** > **Object Management**.

Step 2 Expand the **PKI** node, and choose **Trusted CAs**.

Step 3 Click **Edit** ( ) next to a trusted CA object.

If **View** ( ) appears instead, the configuration belongs to an ancestor domain, or you do not have permission to modify the configuration.

Step 4 Click **Add CRL** and upload your DER or PEM-encoded CRL file.

Step 5 Click **OK** to save the changes.

The certificate revocation list is added to the trusted CA object, updating it with the new CRL information.

What to do next

- If an active policy references your object, deploy configuration changes; see Deploy configuration changes.

External certificate objects

An external certificate object is a certificate management item that

- represents a server public key certificate that does not belong to your organization,
- consists of the object name and certificate, and
- can be used in SSL rules to control traffic that is encrypted with the server certificate. For example, you can upload a self-signed server certificate that you trust, but cannot verify by using a trusted CA certificate.

Supported certificate formats

You can configure an external certificate object by uploading an X.509 v3 server certificate. The supported file formats include:

- Distinguished Encoding Rules (DER)
- Privacy-enhanced Electronic Mail (PEM)

The system validates the file before saving the object, ensuring your certificate is correctly processed. If the certificate is encoded in the PEM format, you can also copy and paste the information.

Add external certificate objects

Add external certificate objects to facilitate secure certificate management and integration with external systems.

Use this task when you need to import and manage external X.509 v3 server certificates in Secure Firewall Management Center for PKI integration.

Procedure

Step 1 Choose **Objects** > **Object Management**.

Step 2 Expand the **PKI** node, and choose **External Certs**.

Step 3 Click **Add External Cert**.

Step 4 Enter a **Name**.

Step 5 Above the **Certificate Data** field, click **Browse** to upload a DER or PEM-encoded X.509 v3 server certificate file.

Step 6 Click **Save**.

What to do next

- If an active policy references your object, deploy configuration changes; see Deploy configuration changes.

Internal certificate objects

An internal certificate object is a server public key certificate representation that

- belongs to your organization and consists of the object name, public key certificate, and paired private key
- supports SSL rule traffic decryption using the known private key, and
- enables authentication for ISE connections and captive portal configurations.

Internal certificate object usage and configuration

You can use internal certificate objects and groups in:

- your SSL rules to decrypt traffic incoming to one of your organization's servers using the known private key.
- your ISE/ISE-PIC connection. Select an internal certificate object for the **MC Server Certificate** field.
- your captive portal configuration to authenticate the identity of your captive portal device when connecting to users' web browsers. Select an internal certificate object for the **Server Certificate** field.

You can configure an internal certificate object by uploading an X.509 v3 RSA-based or elliptic curve-based server certificate and paired private key. You can upload a file in one of these supported formats:

- Distinguished Encoding Rules (DER)
- Privacy-enhanced Electronic Mail (PEM)

If the file is password-protected, you must supply the decryption password. If the certificate and key are encoded in the PEM format, you can also copy and paste the information.

You can upload only files that contain proper certificate or key information, and that are paired with each other. The system validates the pair before saving the object.

After you create the internal certificate object, you can modify the name, but cannot modify other object properties.

You cannot delete an internal certificate object that is in use. Additionally, after you edit an internal certificate object that is in use, the associated access control policy goes out-of-date. You must re-deploy the access control policy for your changes to take effect.

Add internal certificate objects

Add and manage internal certificate objects for secure handling of server certificates and associated private keys within the system.

Use this task when you need to upload and manage DER or PEM-encoded X.509 v3 server certificates and their corresponding private keys for internal use. This is typically required for secure communication and authentication within your environment.

Procedure

Step 1 Choose **Objects** > **Object Management**.

Step 2 Expand the **PKI** node, and choose **Internal Certs**.

Step 3 Click **Add Internal Cert**.

Step 4 Enter a **Name**.

Step 5 Above the **Certificate Data** field, click **Browse** to upload a DER or PEM-encoded X.509 v3 server certificate file.

Step 6 Above the **Key** field, or click **Browse** to upload a DER or PEM-encoded paired private key file.

Step 7 If the uploaded private key file is password-protected, check the **Encrypted, and the password is:** check box, and enter the password.

Step 8 Click **Save**.

Certificate enrollment objects

A certificate enrollment object is a PKI configuration item that

- contains the Certification Authority (CA) server information and enrollment parameters required for creating Certificate Signing Requests (CSRs)
- obtains identity certificates from the specified CA, and
- may include certificate revocation information.

Certificate enrollment object reference information

Trustpoints let you manage and track CAs and certificates. A trustpoint is a representation of a CA or identity pair. A trustpoint includes the identity of the CA, CA-specific configuration parameters, and an association with one, enrolled identity certificate.

This section describes how to use and manage certificate enrollment objects in your PKI infrastructure.

- Certificate enrollment objects are used to enroll managed devices into your PKI infrastructure and create trustpoints (CA objects) on devices that support VPN connections.
- To manage certificate enrollment objects, go to **Objects** > **Object Management** > **PKI** > **Certificate Enrollment**. The following information is shown:
  - Existing certificate enrollment objects are listed in the **Name** column. Use the search field (the magnifying glass) to filter the list.
  - The enrollment type of each object is shown in the **Type** column. These enrollment methods can be used:
    - **Self Signed**—The managed device generates its own self signed root certificate.
    - **EST**—Enrollment over Secure Transport is used by the device to obtain an identity certificate from the CA.
    - **SCEP**—(Default) Simple Certificate Enrollment Protocol (SCEP) is used by the device to obtain an identity certificate from the CA.
    - **Manual**—The process of enrolling is carried out manually by the administrator.
    - **PKCS12 File**—Import a PKCS12 file on a Threat Defense managed device that supports VPN connectivity. A PKCS#12, or PFX or P12 file holds the server certificate, any intermediate certificates, and the private key in one encrypted file. Enter the **Passphrase** value for decryption.
  - The **Override** column indicates whether the object allows overrides (a green check mark) or not (a red X). If a number is displayed, it is the number of overrides in place. Use the Override option to customize the object settings for each device that is part of the VPN configuration. Overriding makes each device's trustpoint details unique. Typically the Common Name or Subject is overridden for each device in the VPN configuration. See Object overrides, on page 12 for details and procedures on overriding objects of any type.
  - **Edit** a previously created certificate enrollment object by clicking on the edit icon (a pencil). Editing can only be done if the enrollment object is not associated with any managed devices. Refer to the adding instructions for editing a certificate enrollment object. Failed enrollment objects can be edited.
  - **Delete** a previously created certificate enrollment object by clicking on the delete icon (a trash can). You cannot delete a certificate enrollment object if it is associated with any managed device.

1. Define parameters for CA authentication and enrollment in a certificate enrollment object. Specify shared parameters and use the override facility to specify unique object settings for different devices.
2. Associate and install this object on each managed device that requires the identity certificate. On the device, it becomes a trustpoint.

   When a certificate enrollment object is associated with and then installed on a device, the process of certificate enrollment starts immediately. The process is automatic for self-signed, SCEP, EST, and

PKCS12 file enrollment types, meaning it does not require any additional administrator action. Manual certificate enrollment requires extra administrator action.

3. Specify the created trustpoint in your VPN configuration.

Press ( )**Add Cert Enrollment** to open the **Add Cert Enrollment** dialog and configure a certificate enrollment object, see Add certificate enrollment objects, on page 71. Then install the certificate on each managed, headend device.

For more information on PKI, digital certificates, and certificate enrollment see PKI infrastructure and digital certificates.

Add certificate enrollment objects

Add certificate enrollment objects to manage device trustpoints and enable secure certificate enrollment for Firewall Threat Defense devices.

- Support multiple enrollment protocols, including SCEP, EST, Manual, and PKCS12 file import.

Use these objects to enable secure certificate management and device trustpoint enrollment. You must have Admin or Network Admin privileges to do this task.

Procedure

Step 1 Open the **Add Certificate Enrollment** dialog:

- Directly from Object Management: In the **Objects** > **Object Management** > **PKI** > **Certificate Enrollment** from the navigation pane, and click **Add Certificate Enrollment**.
- While configuring a managed device: In the **Devices** > **Certificates** screen, choose **Add** > **Add New Certificate** and click ( ) for the **Certificate Enrollment** field.

Step 2 Enter the **Name**. When enrollment is complete, this name is used as the trustpoint name on the managed devices with which it is associated. Click the **CA Information** tab, and then choose the **Enrollment Type**.

- **Self-Signed Certificate**—The managed device, acting as a CA, generates its own self-signed root certificate. No other information is needed in this pane.

Note When enrolling a self-signed certificate you must specify the Common Name (CN) in the certificate parameters.

- **EST**—Enrollment over Secure Transport protocol. Specify the EST information. See EST enrollment configuration options in Certificate Enrollment Object , on page 73.
- **SCEP**—(Default) Simple Certificate Enrollment Protocol. Specify the SCEP information. See Certificate Enrollment Object SCEP options , on page 74.
- **Manual**
  - **CA Only**—Check this check box to create only the CA certificate from the selected CA. An identity certificate will not be created for this certificate.

    If you do not select this check box, a CA certificate is not mandatory. You can generate the CSR without having a CA certificate and obtain the identity certificate.

- **CA Certificate**—Paste the CA certificate in the PEM format in the box. You can also obtain a CA certificate by copying it from another device.

  You can leave this box empty if you choose to generate a CSR without the CA certificate.

- **PKCS12 File**—Import a PKCS12 file on a Firewall Threat Defense managed device that supports VPN connectivity. A PKCS#12, or PFX, file holds a server certificate, intermediate certificates, and a private key in one encrypted file. Enter the **Passphrase** value for decryption.

Step 3 **Skip Check for CA flag in basic constraints of the CA Certificate**—Check this check box if you want to skip checking the basic constraints extension and the CA flag in a trustpoint certificate.

Step 4 **Validation Usage**—Choose from the options to validate the certificate during a VPN connection:

- **IPsec Client**—Validate an IPsec client certificate for a site-to-site VPN connection.
- **SSL Client**—Validate an SSL client certificate during a remote access VPN connection attempt.
- **SSL Server**—Select to validate an SSL server certificate, such as a Cisco Umbrella server certificate.

Step 5 (Optional) Click the **Certificate Parameters** tab and specify the certificate contents. Refer to Certificate Enrollment Object certificate parameters in certificate requests , on page 75.

This information is placed in the certificate and is readable by any party who receives the certificate from the router.

Step 6 (Optional) Click the **Key** tab and specify the Key information. Refer to Certificate Enrollment Object key options , on page 76.

Step 7 (Optional) Click the **Revocation** tab and specify the revocation options: Refer to Certificate Enrollment Object revocation options , on page 77.

Step 8 **Allow Overrides** of this object if desired.

When you allow overrides in the PKCS12 certificate enrollment object, update the**Passphrase** for the certificate on the device where you override it.

See Object overrides, on page 12 for a full description of object overrides.

Step 9 Click **Save**.

What to do next

Associate and install the enrollment object on a device to create a trustpoint on that device.

Related Topics

Install a certificate using self-signed enrollment
Install a certificate using EST enrollment
Install a certificate using SCEP enrollment
Install a certificate using manual enrollment
Install a certificate using a PKCS12 file

Add a certificate enrollment

A valid IdP certificate ensures that your system can securely authenticate users through an identity provider. This is essential for secure communications and compliance.

Procedure

Step 1 Enter the **Name**.

Step 2 Paste the certificate information in the **IdP Certificate** field in PEM format.

Note If the certificate is dependent on a root or intermediate certificate, you must install the dependent certificates. See Certificates.

Step 3 Click **Save**.

EST enrollment configuration options in Certificate Enrollment Object

This topic describes the EST enrollment options and the required fields for configuring EST certificate enrollment in the Secure Firewall Management Center.

Secure Firewall Management Center navigation path

Objects > **Object Management** > **PKI** > **Certificate Enrollment**. Click **Add Certificate Enrollment** to open the **Add Certificate Enrollment** dialog, and select the **CA Information** tab.

Fields

Enrollment Type—set to **EST**.

Note
- EST enrollment type does not support EdDSA key.
- EST's ability to auto-enroll a device when its certificate expires is not supported.

Enrollment URL—Enter the URL of the CA server to which devices should attempt to enroll.

Use an HTTPS URL in the form of **https://CA_name:port**, where *CA_name* is the host DNS name or IP address of the CA server. The *port* number is mandatory.

Username—The username required to access the CA server.

Password / Confirm Password—The password required to access the CA server.

Fingerprint—Optionally, enter the fingerprint of the CA server certificate. Verifying the fingerprint helps ensure you are connecting to the legitimate CA server and not a substitute. Enter the fingerprint in hexadecimal format. If the fingerprint does not match, the certificate is rejected. Obtain the fingerprint directly from the CA server.

Source Interface—The interface that interacts with the CA server. By default, the diagnostic interface is displayed. To configure a data interface as the source interface, choose the respective security zone or interface group object.

Ignore EST Server Certificate Validations—By default, EST server certificate validation is enabled. Check the check box to ignore Firewall Threat Defense validating EST server certificate.

Certificate Enrollment Object SCEP options

This topic describes the SCEP enrollment options and fields available when adding a certificate enrollment object in Secure Firewall Management Center.

Secure Firewall Management Center navigation path

Objects > **Object Management** > **PKI** > **Certificate Enrollment**. Click **Add Certificate Enrollment** to open the **Add Certificate Enrollment** dialog, and select the **CA Information** tab.

Fields

Enrollment Type—Set to **SCEP**.

Enrollment URL—Enter the URL of the CA server to which devices should attempt to enroll.

Use an HTTP URL in the format **`http://CA_name:port,`** . The CA_name is the DNS name or IP address of the CA server, and the port number is mandatory.

Note If the SCEP Server is referred with hostname/FQDN, configure the DNS Server using FlexConfig object.

If the CA cgi-bin script location at the CA is not the default (/cgi-bin/pkiclient.exe) include the nonstandard script location in the URL, in the form of http://CA_name:port/script_location, where script_location is the full path to the CA scripts.

Challenge Password / Confirm Password—The password used by the CA server to validate the identity of the device. You can obtain the password by contacting the CA server directly or by entering the following address in a web browser: **`http://URLHostName/certsrv/mscep/mscep.dll`**. Because the password is valid for only 60 minutes, deploy it as soon as possible after creation.

Retry Period—The interval (in minutes) between certificate request attempts. Values range from 1 to 60 minutes. The default is 1 minute.

Retry Count—The number of retries that should be made if no certificate is issued on the first request. Value can be 1 to 100. The default is 10.

CA Certificate Source—Specifies how the CA certificate will be obtained.

- **Retrieve Using SCEP** (default and only supported option)—Retrieve the certificate from the CA server using the Simple Certificate Enrollment Process (SCEP). Using SCEP requires a connection between your device and the CA server. Ensure a network route exists between your device and the CA server before beginning enrollment.

Fingerprint—When retrieving the CA certificate using SCEP, you may enter the fingerprint for the CA server. Using the fingerprint to verify the authenticity of the CA server's certificate helps prevent an unauthorized party from substituting a fake certificate in place of the real one. Enter the **Fingerprint** for the CA server in hexadecimal format. If the value you enter does not match the fingerprint on the certificate, the certificate is rejected. Obtain the CA's fingerprint by contacting the server directly, or by opening: **`http://<URLHostName>/certsrv/mscep/mscep.dll`**.

Certificate Enrollment Object certificate parameters in certificate requests

When sending a certificate request to a CA server, you can specify additional information that is placed into the certificate and is viewable by recipients. All information should follow the standard LDAP X.500 format.

Secure Firewall Management Center navigation path

Objects > **Object Management** > **PKI** > **Certificate Enrollment**. Click **Add Certificate Enrollment** to open the **Add Certificate Enrollment** dialog box, and select the **Certificate Parameters** tab.

Fields

Field	Description
Include FQDN	Whether to include the device's fully qualified domain name (FQDN) in the certificate request. You can select from these options: • **Use Device Hostname as FQDN** • **Don't use FQDN in certificate** • **Custom FQDN**—Select this and then specify it in the **Custom FQDN** field that displays.
Include Device's IP Address	Specify the interface whose IP address you want to include in the certificate request.
Common Name (CN)	Specify the X.500 common name to include in the certificate. **Note** When enrolling a self-signed certificate you must specify the Common Name (CN) in the certificate parameters.
Organization Unit (OU)	Specify the name of the organization unit (for example, a department name) to include in the certificate.
Organization (O)	Specify the organization or company name to include in the certificate.
Locality (L)	Specify the state or province to include in the certificate.
State (ST)	The state or province to include in the certificate.
County Code (C)	The country to include in the certificate. These codes conform to ISO 3166 country abbreviations, for example "US" for the United States of America.
Email (E)	Specify the email address to include in the certificate.
Include Device's Serial Number	Specify whether to include the device's serial number in the certificate. The CA uses the serial number to authenticate certificates or to later associate a certificate with a particular device. If you are unsure, include the serial number, because it is useful for debugging purposes.

Certificate Enrollment Object key options

This topic describes the available key options and their settings when adding certificate enrollment and selecting the Key tab in the Certificate Enrollment Object dialog box.

Secure Firewall Management Center navigation path

Objects > **Object Management** > **PKI** > **Certificate Enrollment**, and click **Add Certificate Enrollment** to open the **Add Certificate Enrollment** dialog box, and select the **Key** tab.

Fields

- **Key Type**: Choose RSA, ECDSA, EdDSA.

Note

- EdDSA is not supported for EST enrollment type.
- EdDSA is supported only in Site-to-Site VPN topologies.
- EdDSA is not supported as an identity certificate for Remote Access VPN.

- **Key Name**: Specifies the name of the key pair. If the key pair already exists, this field refers to the existing name. If not, it defines the name for a new key pair to be generated during enrollment. If no name is specified, the fully qualified domain name (FQDN) key pair is used instead.
- **Key Size**: When generating a new key pair, defines the desired key size (modulus) in bits. The recommended size is 2048 bits. Larger modulus sizes offer more security but require more time (a minute or more when larger than 512 bits) to generate and process.

Important

- On Firewall Management Center and Firewall Threat Defense Versions 7.0 and higher, you cannot enroll certificates with RSA key sizes smaller than 2048 bits and keys using SHA-1 with the RSA Encryption algorithm.
- You can use the PKI Enrollment of Certificates with Weak-Crypto to allow certificates with SHA-1 and smaller key size.
- You cannot generate RSA keys with sizes smaller than 2048 bits for Firewall Threat Defense 7.0, even when you enable the weak-crypto option.

- **Advanced Settings**: Select **Ignore IPsec Key Usage** if you do not want to validate values in the key usage and extended key usage extensions of IPsec remote client certificates. You can suppress key usage checking on IPsec client certificates. By default, this option is not enabled.

Note For site-to-site VPN connection, if you use a Windows Certificate Authority (CA), the default Application Policies extension is **IP security IKE intermediate**. If you are using this default setting, you must select the **Ignore IPsec Key Usage** option for the object you select. Otherwise, the endpoints cannot complete the site-to-site VPN connection.

Enroll PKI certificates with weak-crypto

Certificates that use weak cryptography (such as SHA-1 signatures or RSA keys smaller than 2048 bits) have restricted support on Firewall Management Center and Firewall Threat Defense devices. This reference details the conditions under which weak-crypto certificates may be enrolled and the required steps for enabling their use.

SHA-1 hashing signature algorithm, and RSA key sizes that are smaller than 2048 bits for certification are not supported on Firewall Management Center and Firewall Threat Defense Version 7.0 and later. You cannot enroll certificates with RSA key sizes that are smaller than 2048 bits.

To override these restrictions on Firewall Management Center 7.0 managing Firewall Threat Defenses running Versions earlier than 7.0, use the enable weak-crypto option on the Firewall Threat Defense. Do not permit weak-crypto keys, because, such keys are not as secure as the ones with larger key sizes.

Note Firewall Threat Defense 7.0 or later does not support generating RSA keys with sizes smaller than 2048 bits even when you permit weak-crypto.

To enable weak-crypto on the device, navigate to the **Devices** > **Certificates** page. Click the **Enable Weak-Crypto**( ) button provided against the Firewall Threat Defense device. When the weak-crypto option is enabled, the button changes to . By default, the weak-crypto option is disabled.

Note When a certificate enrollment fails due to weak cipher usage, the Firewall Management Center displays a warning message prompting you to enable the weak-crypto option. Similarly, when you turn on the enable weak-crypto button, the Firewall Management Center displays a warning message before enabling weak-crypto configuration on the device.

Upgrade earlier versions to Firewall Threat Defense 7.0

When you upgrade to Firewall Threat Defense 7.0, the existing certificate configurations are retained. However, if those certificates have RSA keys smaller than 2048 bits and use SHA-1 encryption algorithm, they cannot be used to establish VPN connections. You must either procure a certificate with RSA key size of at least 2048 bits or enable the permit weak-crypto option for VPN connections.

Certificate Enrollment Object revocation options

Specify whether to check the revocation status of a certificate by choosing and configuring the method. Revocation checking is off by default. The system does not check either method (CRL or OCSP).

Secure Firewall Management Center navigation path

Objects > **Object Management** > **PKI** > **Certificate Enrollment**, and click ( ) **Add Cert Enrollment** to open the **Add Cert Enrollment** dialog, and select the **Revocation** tab.

Fields

- **Enable Certificate Revocation Lists**—Check to enable CRL checking.

- **Use CRL distribution point from the certificate**—Check to obtain the revocation lists distribution URL from the certificate.
- **Use static URL configured**—Check this to add a static, predefined distribution URL for revocation lists. Then add the URLs.

  CRL Server URLs—Specify the URL of the LDAP server from which the CRL can be downloaded.

  URLs must start with **`http://`**. Include a port number in the URL. Enclose IPv6 addresses in square brackets, for example: *http://[0:0:0:0:0.18:0a01:7c16]*.

- **Enable Online Certificate Status Protocol (OCSP)**—Check to enable OCSP checking.

  OCSP Server URL—The URL of the OCSP server checking for revocation if you require OCSP checks.

  URLs must start with **`http://`**. Enclose IPv6 addresses in square brackets, for example: *http://[0:0:0:0:0.18:0a01:7c16]*.

- **Consider the certificate valid if revocation information cannot be reached**—Checked by default. Uncheck if you do not want to allow this.

Configure a policy list

Configure a policy list to group and manage multiple match statements for use in route maps. This allows for efficient policy management and flexible routing decisions.

Use the Configure Policy List page to create, copy, and edit policy list objects. You can create policy list objects to use when you are configuring route maps. When a policy list is referenced within a route map, all of the match statements in that policy list are evaluated and processed. With a route map, you can configure two or more policy lists. A policy list can coexist with other match and set statements that are configured in the same route map but outside the policy list. When multiple policy lists perform matching within a route map entry, only the incoming attribute is matched by all policy lists.

You can use this object with Firewall Threat Defense devices.

Procedure

Step 1 Select **Objects** > **Object Management** > **Policy List**.

Step 2 Click **Add Policy List**. Specify these values:

a. Enter a name for the policy list object in the **Name** field. Object names are not case-sensitive.

b. Select whether to allow or block access for matching conditions from the **Action** drop-down list.

Step 3 Enter the details in each of these tabs:

Tabs	Details
Interface	Use this tab to distribute routes that have their next hop out of one of the interfaces specified. In the **Zones/Interfaces** list, add the zones that contain the interfaces through which the device communicates with the management station. For interfaces not in a zone, you can type the interface name into the field below the **Selected Zone/Interface** list and click **Add**. The host will be configured on a device only if the device includes the selected interfaces or zones.
Address	Use this tab to redistribute any routes that have a destination address that is permitted by a standard access list or prefix list. Choose whether to use an **Access List** or **Prefix List** for matching and then enter or select the Standard Access List Objects or Prefix list objects you want to use for matching.
Next Hop	Use this tab to redistribute any routes that have a next hop router address passed by one of the access lists or prefix lists specified. Choose whether to use an **Access List** or **Prefix List** for matching and then enter or select the Standard Access List Objects or Prefix list objects you want to use for matching.
Route Source	Use this tab to redistribute routes that have been advertised by routers and access servers at the address specified by the access lists or prefix list. Choose whether to use an **Access List** or **Prefix List** for matching and then enter or select the Standard Access List Objects or Prefix list objects you want to use for matching.
AS Path	Use this tab to match a BGP autonomous system path. If you specify more than one AS path, then the route can match either AS path.
Community Rule	Use this tab to enable matching of the BGP community or extended community with the specified community list objects or the extended community list objects respectively. If you specify more than one rule, the routes are verified against the rules until a matching permit or deny is met. **a.** To specify a community list to the rule, click **Edit** (✎) given in the **Selected Community List** field. The community lists appear under **Available Community List**. Select the required list, click **Add**, and then click **Ok**. To enable matching the BGP community exactly with the specified community, check the **Match the specified community exactly** check box. **b.** To add the extended community list, click **Edit** (✎) given in the **Selected Extended Community List** field. The extended community lists appear under the **Available Extended Community List**. Select the required list, click **Add**, and then click **Ok**. **Note** The extended community lists are applicable only for configuring import or export of routes.

Tabs	Details
Metric & tag	Use this tab to match the metric and security group tag of a route. **a.** Enter the metric values to use for matching in the **Metric** field. You can enter multiple values, separated by commas. This setting allows you to match any routes that have a specified metric. The metric values can range from 0 to 4294967295. **b.** Enter the tag values to use for matching in the **Tag** field. You can enter multiple values separated by commas. This setting allows you to match any routes that have a specified security group tag. The tag values can range from 0 to 4294967295.

Step 4 If you want to allow overrides for this object, check the **Allow Overrides** check box; see Allow object overrides, on page 14.

Step 5 Click **Save**.

Ports

A port object is a network configuration element that

- represents different protocols, such as TCP, UDP, ICMP, and others, in distinct ways
- can specify protocol numbers, port numbers or ranges, and optional type and code for certain protocols, and
- is used to define access and monitoring rules within the system's web interface.

Port object representations and usage

Port objects represent protocols differently depending on their type.

- **TCP and UDP**: Port objects represent these transport layer protocols by specifying the protocol number, and may include an associated port or port range. For example: `TCP(6)/22`.
- **ICMP and ICMPv6 (IPv6-ICMP)**: Port objects represent these Internet layer protocols by specifying the protocol number and optional type and code. For example: `ICMP(1):3:3`.

  Restrict an ICMP or IPV6-ICMP port object by type and, if needed, code. Learn more about ICMP types and codes at:

  - ICMP Types and Codes: http://www.iana.org/assignments/ICMP-parameters/ICMP-parameters.xml
  - ICMPv6 Types and Codes: http://www.iana.org/assignments/icmpv6-parameters/icmpv6-parameters.xml

- **Other protocols**: Port objects can represent protocols that do not use ports.

The system provides default port objects for well-known ports. Default objects are read-only and always available for use. You can create custom port objects or groups in addition to the default ones.

Port objects and groups can be used throughout the system's web interface in access control policies, identity rules, network discovery rules, port variables, and event searches. For example, you can configure a network

discovery policy to exclude monitoring ports used by a custom client if it generates excessive or misleading events.

Guidelines for using port objects

- You cannot add any protocol other than TCP or UDP for source port conditions in access control rules.
- Set source and destination port conditions in a rule using only one transport protocol.
- Add only supported protocols to a port object group used in a source port condition so that the rule takes effect on the managed device when you deploy the configuration.
- To add a destination port, create a port object using only TCP or only UDP ports for the source port condition in a rule. Similarly, to add a source port, create a port object using only TCP or only UDP ports as the destination port condition.

Create port objects

Create port objects to specify protocol and port information for network traffic management in Secure Firewall Management Center.

Use this task to add or modify port objects in access control or other security policies within Secure Firewall Management Center. Port objects allow you to specify which network traffic is allowed or blocked based on protocol and port criteria. They are critical when you set up or update security rules.

Procedure

Step 1 Choose **Objects** > **Object Management**.

Step 2 Choose **Port** from the object types list.

Step 3 Choose **Add Object** from the **Add Port** drop-down list.

Alternatively, clone an existing port object and edit its parameters to create a new port object. Click the **Clone** icon on the existing port object that you want to clone.

Step 4 Enter a **Name**.

Step 5 Choose a **Protocol**.

Step 6 Depending on the protocol you chose, constrain by **Port**, or choose an ICMP **Type** and **Code**.

You can enter ports from `1` to `65535`. Use a hyphen to specify a port range. You must constrain the object by port if you chose to match **All** protocols, using the **Other** drop-down list.

Step 7 Manage overrides for the object:

- If you want to allow overrides for this object, check the **Allow Overrides** check box; see Allow object overrides, on page 14.
- If you want to add override values to this object, expand the Override section and click **Add**; see Add object overrides, on page 14.

Step 8 Click **Save**.

What to do next

- If an active policy references your object, deploy configuration changes; see Deploy configuration changes.

Import port objects

To import port objects, refer to Import objects, on page 4.

Prefix lists

A prefix list is a route filtering object that

- defines IPv4 and IPv6 address ranges
- is used when configuring route maps, policy maps, OSPF filtering, or BGP neighbor filtering, and
- enables control over which routes are permitted or denied.

Configure an IPv6 prefix list

Use IPv6 prefix lists in Firewall Threat Defense devices to specify which IPv6 routes are allowed or blocked during redistribution. This enhances routing policy control for protocols such as OSPF and BGP.

Procedure

Step 1 Select **Objects** > **Object Management** > **Prefix Lists** > **IPv6 Prefix List** from the table of contents.

Step 2 Click **Add Prefix List**.

a. Enter a name for the prefix list object in the **Name** field

b. Click **Add**.

Step 3 Select the appropriate action, Allow or Block, from the **Action** drop-down list to indicate the redistribution access.

Step 4 Enter a unique number that indicates the position a new prefix list entry will have in the list of prefix list entries already configured for this object, in the **Sequence No.** field. If left blank, the sequence number will default to five more than the largest sequence number currently in use.

Step 5 Specify the IPv6 address in the IP address/mask length format in the **IP address** field. The mask length must be a valid value between 1 and 128.

Step 6 Enter the minimum prefix length in the **Minimum Prefix Length** field and the maximum prefix length in the **Maximum Prefix Length** field.

The minimum prefix length value must be greater than the mask length and less than or equal to the Maximum Prefix Length, if specified.

If the Minimum Prefix Length is present, the maximum prefix length value must be greater than or equal to it. If the Minimum Prefix Length is not specified, the maximum value must be greater than the mask length.

Step 7 Click **Add**.

Step 8 If you want to allow overrides for this object, check the **Allow Overrides** check box; see Allow object overrides, on page 14.

Step 9 Click **Save**.

Configure an IPv4 prefix list

Use the Configure IPv4 Prefix list page to manage prefix list objects. These objects can be used when configuring route maps, policy maps, OSPF filtering, or BGP neighbor filtering.

You can use this object with Firewall Threat Defense devices.

Procedure

Step 1 Select **Objects** > **Object Management** > **Prefix Lists** > **IPv4 Prefix List** from the table of contents.

Step 2 Click **Add Prefix List**.

a. Enter a name for the prefix list object in the **Name** field

b. Click **Add**.

Step 3 Select the appropriate action, Allow or Block, from the **Action** drop-down list to indicate the redistribution access.

Step 4 Enter a unique number in the **Sequence No.** field to specify the position of the new prefix list entry. If the field is left blank, the system assigns a sequence number that is five higher than the largest one currently in use.

Step 5 Specify the IPv4 address in the IP address/mask length format in the **IP address** field. The mask length must be a valid value between 1 and 32.

Step 6 Enter the minimum prefix length in the **Minimum Prefix Length** field and the maximum prefix length in the **Maximum Prefix Length** field.

The minimum prefix length value must be greater than the mask length and less than or equal to the Maximum Prefix Length, if specified.

If the Minimum Prefix Length is present, the maximum prefix length value must be greater than or equal to it. If the Minimum Prefix Length is not specified, the maximum value must be greater than the mask length.

Step 7 Click **Add**.

Step 8 If you want to allow overrides for this object, check the **Allow Overrides** check box; see Allow object overrides, on page 14.

Step 9 Click **Save**.

Configure a route map

Route maps are used to manage route redistribution between routing protocols and to control default route generation. Route maps can specify which routes are permitted or denied and apply rules for matching route attributes.

You can use this object with Firewall Threat Defense devices.

Configure a route map to create or edit route map entries in a Route Map object.

Before you begin

A Route Map can use one or more of these objects. It is not mandatory to add all of them. Create and use any of these objects as required to configure your route map.

- Add ACLs.
- Add Prefix Lists.
- Add AS Path.
- Add Community Lists.
- Add Extended Community Lists.



Note The extended community lists apply only to configuring import or export of routes.

- Add Policy Lists.

Procedure

Step 1 Select **Objects** > **Object Management** > **Route Map**.

Step 2 Click **Add Route Map**.

Step 3 Click **Add** on the **New Route Map Object** window.

Step 4 In the **Sequence No.** field, enter a number, from 0 through 65535, that indicates the position a new route map entry has in the list of route maps entries already configured for this route map object.



Note We recommend that you number clauses in intervals of at least 10 to reserve numbering space in case you want to insert clauses in the future.

Step 5 Select Allow or Block from the **Redistribution** drop-down list, to indicate the redistribution access.

Step 6 Click the **Match Clauses** tab to match routes or traffic based on the following criteria, which you select in the table of contents:

- **Security Zones** —Match traffic based on the (ingress/egress) interfaces. You can select zones and add them, or type in interface names and add them.
- **IPv4**— Match IPv4 (routes/traffic) based on the following criteria; select the tab to define the criteria.
  a. Click the **Address** tab to match routes based on the route address. For IPv4 addresses, choose whether to use an Access list or Prefix list for matching from the drop-down list and then enter or select the ACL objects or Prefix list objects you want to use for matching.
  b. Click the **Next Hop** tab to match routes based on the next hop address of a route. For IPv4 addresses, choose whether to use an access list or Prefix list for matching from the drop-down list and then enter or select the ACL objects or Prefix list objects you want to use for matching.
  c. Click the **Route Source** tab to match routes based on the advertising source address of the route. For IPv4 addresses, choose whether to use an access list or Prefix list for matching from the drop-down list and then enter or select the ACL objects or Prefix list objects you want to use for matching.
- **IPv6** —Match IPv6 (routes/traffic) based on the route address, next-hop address or advertising source address of route.
- **BGP**—Match BGP (routes/traffic) based on the following criteria; select the tab to define the criteria.
  a. Click the **AS Path** tab to enable matching the BGP autonomous system path access list with the specified path access list. If you specify more than one path access list, then the route can match either path access list.
  b. Click the **Community List** tab to enable matching of the BGP community or extended community with the specified community list objects or the extended community list objects respectively.
     - To specify a community list to the rule, click **Edit** ( ) given in the **Selected Community List** field. The community lists appears under **Available Community List**. Select the required list, click **Add**, and then click **Ok**. For information on how to create community list objects, see Configure a community list, on page 30
     - To add the extended community list, click **Edit** ( ) given in the **Selected Extended Community List** field. The extended community lists appears under the **Available Extended Community List**. Select the required list, click **Add**, and then click **Ok**. For information on how to create extended community list objects, see Configure an extended community list, on page 31.

     To enable matching the BGP community exactly with the specified community list objects, check the **Match the specified community exactly** check box. This option is not applicable for the extended community list.

Note If you specify more than one rule, the routes are verified against the rules until a matching permit or deny condition is met. Any route that does not match at least one Match community will not be advertised for outbound route maps.

  c. Click the **Policy List** tab to configure a route map to evaluate and process a BGP policy. When multiple policy lists perform matching within a route map entry, all policy lists match on the incoming attribute only.

- **Others** —Match routes or traffic based on the following criteria.

a. Enter the metric values to use for matching in the **Metric Route Value** field, to enable matching the metric of a route. You can enter multiple values separated by commas. This setting allows you to match any routes that have a specified metric. The metric values can range from 0 to 4294967295.

b. Enter the tag values to use for matching in the **Tag Values** field. You can enter multiple values separated by commas. This setting allows you to match any routes that have a specified security group tag. The tag values can range from 0 to 4294967295.

c. Check the appropriate **Route Type** option to enable matching of the route type. Valid route types are External1, External2, Internal, Local, NSSA-External1, and NSSA-External2. You can choose more than one route type from the list.

Step 7 Click the **Set Clauses** tab to set routes/traffic based on the following criteria, which you select in the table of contents:

- **Metric Values**—Set either Bandwidth, all of the values or none of the values.
  a. Enter a metric value or bandwidth in Kbits per second in the **Bandwidth** field. Valid values are an integer value in the range from 0 to 4294967295.
  b. Select to specify the type of metric for the destination routing protocol, from the **Metric Type** drop-down list. Valid values are: internal, type-1, or type-2.

- **BGP Clauses** —Set BGP routes based on the following criteria; select the tab to define the criteria.
  a. Click the **AS Path** tab to modify an autonomous system path for BGP routes.
     1. Enter an AS path number in the **Prepend AS Path** field to prepend an arbitrary autonomous system path string to BGP routes. Usually the local AS number is prepended multiple times, increasing the autonomous system path length. If you specify more than one AS path number then the route can prepend either AS number.
     2. Enter an AS path number in the **Prepend Last AS to AS Path** field to prepend the AS path with the last AS number. Enter a value for the AS number from 1 to 10.
     3. Check the **Convert route tag into AS path** check box to convert the tag of a route into an autonomous system path.

  b. Click the **Community List** tab to set the community attributes:

     Under **Specific Community**:

     1. Click the **None** radio button, to remove the community attribute from the prefixes that pass the route map.
     2. Click the **Specific Community** radio button, to enter a community number, if applicable. Valid values are from 1 to 4294967295.
     3. Check the **Add to existing communities** check box, to add the community to the already existing communities.
     4. Select the **Internet**, **No-Advertise**, or **No-Export** check-boxes to use one of the well-known communities.

     Under **Specific Extended Community**, in the **Route Target** field, enter the route target number in *ASN:nn* format:

- You can enter values that ranges from 1:1 to 65534:65535.

  You can add a single route target or a set of route targets separated by commas in a single entry. For example, *1:2,1:4,1:6*.

- You can have a maximum of 8 route targets in an entry.
- You cannot have redundant route target entries across route maps.

c. Click the **Others** tab to set additional attributes.

1. Check the **Set Automatic Tag** check-box to automatically compute the tag value.
2. Enter a preference value for the autonomous system path in the **Set Local Preference** field. Enter a value between 0 and 4294967295.
3. Enter a BGP weight for the routing table in the **Set Weight** field. Enter a value between 0 and 65535.
4. Select to specify the BGP origin code. Valid values are **Local IGP** and **Incomplete**.
5. In the IPv4 Settings section, specify a next hop IPv4 address of the next hop to which packets are output. It need not be an adjacent router. If you specify more than one IPv4 address then the packets can output at either IP address.

   Select to specify an IPv4 prefix list in the **Prefix List** drop-down list.
6. In the IPv6 Settings section, specify a next hop IPv6 address of the next hop to which packets are output. It need not be an adjacent router. If you specify more than one IPv6 address, then the packets can output at any of the IP addresses.

   Select to specify an IPv6 prefix in the **Prefix List** drop-down list.

Step 8 Click **Add**.

If you want to allow overrides for this object, check the **Allow Overrides** check box; see Allow object overrides, on page 14.

Step 9 Click **Save**.

After completing these steps, the route map is configured and ready for use in route redistribution and policy enforcement. The route map entries and associated match and set clauses are applied AS specified.

What to do next

Review the route map configuration to ensure that all required match and set clauses are correctly defined. Make any necessary adjustments to optimize routing policies.

- Monitor route redistribution and verify that routes are processed according to the configured route map.

Security intelligence

Security Intelligence is a traffic filtering mechanism that

- uses collections of IP addresses, domain names, and URLs to quickly filter traffic that matches entries
- requires the IPS license (for Firewall Threat Defense devices) or the Protection license (all other device types), and
- operates through lists and feeds that are grouped into DNS, Network, and URL categories.

Security intelligence components

Security Intelligence *lists* and *feeds* are collections of IP addresses, domain names. You can use these to quickly filter traffic that matches an entry on a list or feed.

- A list is a static collection that you manage manually.
- A feed is a dynamic collection that updates on an interval over HTTP or HTTPS.

Security Intelligence lists/feeds are grouped into:

- DNS (Domain names)
- Network (IP addresses)
- URLs

System-provided feeds:

Cisco provides these feeds as Security Intelligence objects:

- Security Intelligence feeds updated regularly with the latest threat intelligence from Talos:
  - Cisco-DNS-and-URL-Intelligence-Feed (under DNS Lists and Feeds)
  - Cisco-Intelligence-Feed (for IP addresses, under Network Lists and Feeds)

  You cannot delete the system-provided feeds, but you can change the frequency of (or disable) their updates.
- Cisco-TID-Feed (under Network Lists and Feeds)

  This feed is not used in the Security Intelligence tab of the access control policy.

  Instead, you must enable and configure Secure Firewall Threat Intelligence Director to use this feed. The feed is a collection of TID observables data.

  Use this object to configure the frequency of data publication to TID elements.

  For more information, see Threat Intelligence Director.

Predefined global Block and Do Not Block lists:

The system includes predefined global Block lists and Do Not Block lists for domains (DNS), IP addresses (Networks), and URLs.

These lists are empty until you populate them. To build these lists, see Global and domain security intelligence lists, on page 89.

By default, access control and DNS policies use these lists as part of Security Intelligence.

Custom feeds:

You can use third-party feeds or use a custom internal feed to easily maintain an enterprise-wide Block list in a large deployment with multiple Secure Firewall Management Center appliances.

See Custom security intelligence feeds, on page 97.

Custom lists:

Custom lists can augment and fine-tune feeds and the global lists.

See Custom Security Intelligence lists, on page 99.

Security Intelligence lists and feeds usage locations:

- IP address and address blocks—Use Block and Do Not Block lists in access control policies, as part of Security Intelligence.
- Domain Names—Use Block and Do Not Block lists in DNS policies, as part of Security Intelligence.
- URLs—Use Block and Do Not Block lists in access control policies, as part of Security Intelligence. You can also use URL lists in access control and QoS rules, whose analysis and traffic handling phases occur after Security Intelligence.

Modify security intelligence objects

This topic explains how to add or delete entries on a Block list, Do Not Block list, feed, or sinkhole object. It also describes the edit capabilities and redeployment requirements for each object type.

The table summarizes the editing options and redeployment requirements for various security intelligence object types.

Object Type	Edit Capabilities	Requires Redeploy After Edit?
Custom Block and Do Not Block lists	Upload new and replacement lists using the object manager.	No
Default (but custom-populated) Block lists and Do Not Block lists: Global, descendant, and domain-specific	Add entries using the context menu or delete entries using the object manager.	No
System-provided Intelligence Feeds	Disable or change update frequency using the object manager.	No
Custom feeds	Fully modify using the object manager.	No
Sinkhole	Fully modify using the object manager.	Yes

Global and domain security intelligence lists

A security intelligence list helps you manage firewall policies by

- lets you manage lists of connections to block or exempt,

- applies your policies consistently across global and domain contexts, and
- enables further threat evaluation without the need to redeploy firewall policies.

Security intelligence list reference information

The Firewall Management Center includes Global Block and Do Not Block lists, so you can use Security Intelligence to consistently block or exempt connections, allowing evaluation by other threat detection processes you configure.

Access control and DNS policies use these Global lists by default, and they apply to all security zones. If you want, you can turn these lists off for individual policies.

- Global Block list: Lets you block specified connections in all security zones.
- Do-Not-Block list: Lets you exempt specified connections from blocking so you can further evaluate threats.



Note These options apply only to Security Intelligence. Security Intelligence does not block traffic already fastpathed. Adding an item to a Security Intelligence Do Not Block list does not mean matching traffic is trusted or fastpathed. For more information, see Security intelligence.

Global and domain security intelligence lists example

For example, if you notice a set of routable IP addresses in intrusion events associated with exploit attempts, you can immediately block those IP addresses. Although it may take a few minutes for your changes to propagate, you do not have to redeploy.

Security intelligence lists and multitenancy

A security intelligence list is a multitenancy feature that

- enables the creation of block or do not block lists that apply to specific subdomains,
- supports descendant domain lists that aggregate the lists of a domain's descendants, and
- allows administrators at or above a domain to populate lists, while only domain-specific administrators can remove items.

Domain lists and descendant domain lists

Domain lists are block or do not block lists whose contents apply to a particular subdomain. The global lists are domain lists for the global domain. Each subdomain has its own named lists, and their contents apply only to that subdomain.

For example, a subdomain named Company A owns these lists:

- Domain Block list - Company A and Domain Do Not Block list - Company A
- Domain Block list for DNS - Company A, Domain Do Not Block list for DNS - Company A
- Domain Block list for URL - Company A, Domain Do Not Block list for URL - Company A

Any administrator at or above the current domain can populate these lists. You can use the context menu to add an item to the block or do not block list in the current and all descendant domains. However, only an administrator in the associated domain can remove an item from a domain list.

For example, a global administrator could add the same IP address to the block list in the global domain and Company A's domain, but not in Company B's domain. This action would add the same IP address to:

- Global Block list (removable only by global administrators)
- Domain Block list - Company A (removable only by Company A administrators)

A descendant domain list is a do not block list or block list that aggregates the domain lists of the current domain's descendants. Leaf domains do not have descendant domain lists.

Descendant domain lists are useful because a higher-level domain administrator can enforce general security intelligence settings, while subdomain users can add items to a block or do not block list in their own deployment.

For example, the global domain has these descendant domain lists:

- Descendant Block lists - Global, Descendant Do Not Block lists - Global
- Descendant Block lists for DNS - Global, Descendant Do Not Block lists for DNS - Global
- Descendant Block lists for URL - Global, Descendant Do Not Block lists for URL - Global



Note Descendant domain lists do not appear in the object manager because they are symbolic aggregations, not hand-populated lists. They appear where you can use them: in access control and DNS policies.

Add entries to global security intelligence lists

Add entries to global Security Intelligence lists to control access and improve threat detection by blocking or exempting specific IP addresses, domains, and URLs.

When reviewing events and dashboards, you can instantly block future traffic involving IP addresses, domains, and URLs that appear in those events by adding them to a predefined Block list.

Similarly, if Security Intelligence is blocking traffic that you want evaluated by threat detection processes subsequent to Security Intelligence blocking, you can add IP addresses, domains, and URLs from events to a predefined Do-Not-Block list.

Traffic is evaluated against entries on these lists during the Security Intelligence phase of threat detection.

For more information about these lists, see Global and domain security intelligence lists, on page 89.

Before you begin

Because adding an entry to a Security Intelligence list affects access control, you must have one of the following user roles:

- Administrator
- Network Admin or Access Admin, plus Security Analyst and Security Approver
- A custom role with both "Modify Access Control Policy" and "Deploy Configuration to Devices" permissions

- Verify that these lists are used in the policies where you expect them to be effective.

Procedure

Step 1 Navigate to an event that includes an IP address, domain, or URL that you want to always block using Security Intelligence, or exempt from Security Intelligence blocking.

Step 2 Right-click the IP address, domain, or URL and choose the appropriate option.

Item Type	Context Menu Option
IP address	Add IP to Block List Add IP to Do-Not-Block List These options add the IP address to the respective lists for Networks.
URL	Add URL to Global Block List for URL Add URL to Global Do-Not-Block List for URL
Domain of a URL in the URL field	Add Domain to Global Block List for URL Add Domain to Global Do-Not-Block List for URL
Domain in the DNS Query field	Add Domain to Global Block List for DNS Add Domain to Global Do-Not-Block List for DNS

What to do next

You do not need to redeploy for these changes to take effect.

If you want to delete an item from a list, see Delete entries from global security intelligence lists, on page 92.

Delete entries from global security intelligence lists

Update your security intelligence lists by removing entries that should no longer be blocked or allowed. This helps you maintain accurate and effective security intelligence lists.

Note To add entries to these lists, see Add entries to global security intelligence lists, on page 91.

Procedure

Step 1 Choose **Objects** > **Object Management** > **Security Intelligence**.

Step 2 Click the appropriate option:

- **Network Lists and Feeds** (for IP addresses)
- **DNS Lists and Feeds** (for domain names)
- **URL Lists and Feeds**

Step 3 Click the pencil beside the Global Block or Global Do-Not-Block list.

Step 4 Click the trash button beside the entry to delete.

List and feed updates for security intelligence feeds

A list and feed update is a security intelligence mechanism that performs the following functions:

- Replaces the existing list or feed file with the contents of the new file.
- Does not merge contents of existing and new files.
- If a corrupt or unrecognizable feed is downloaded, the system keeps using previous feed data. This is not true for the first download. If the system recognizes at least one entry in the feed, it uses only the entries it can recognize.

Feed update frequency and device polling

- Each feed updates the Firewall Management Center every two hours by default; you can modify this frequency.
- When the Firewall Management Center receives updates, it immediately passes them to your managed devices.
- Your managed devices poll the Firewall Management Center every 30 minutes for changes. You cannot change this polling frequency.
- For Secure Firewall 200, updates are downloaded once a day, not every 30 minutes.

To modify feed update intervals, see Change the update frequency for security intelligence feeds, on page 93.

Change the update frequency for security intelligence feeds

Set the intervals at which Security Intelligence feeds are updated in the system. Adjusting the update frequency ensures that threat data is refreshed according to your organization's needs.

You can specify the intervals at which the Firewall Management Center updates Security Intelligence Feeds.

For details about feed updates, see List and feed updates for security intelligence feeds, on page 93.

Before you begin

No explicit prerequisites are stated. Ensure you have access to the system interface where Security Intelligence feeds are managed.

Follow these steps to change the update frequency for Security Intelligence feeds:

Procedure

Step 1 Choose **Objects** > **Object Management**.

Step 2 Expand the **Security Intelligence** node, then choose the feed type whose frequency you want to change.

The system-provided URL feed is combined with the domain feed under **DNS Lists and Feeds**.

Step 3 Next to the feed you want to update, click **Edit** ( ).

If **View** ( ) appears instead, the object belongs to an ancestor domain, or you do not have permission to modify the object.

Step 4 Edit the **Update Frequency**.

Step 5 Click **Save**.

The Security Intelligence feed will update at the new interval you specified. The system will retrieve threat data according to the configured frequency.

Custom security intelligence lists and feeds

This reference describes custom Security Intelligence lists and feeds.

List custom lists and feeds requirements

Each custom list or feed must be a simple text file no larger than 500 MB. List files must have the `.txt` extension. Include one entry or comment per line—for example, one IP address, one URL, or one domain name per line.

List and feed formatting

Each custom list or feed must be a simple text file no larger than 500 MB. List files must have the `.txt` extension. Include one entry or comment per line—for example, one IP address, one URL, or one domain name per line.

Tip The number of entries is limited by the maximum file size. For example, a URL list with no comments and an average URL length of 100 characters (including Unicode representations and newlines) can support more than 5.24 million entries.

- In a DNS list entry, you can use an asterisk (`*`) wildcard character for a domain label. All labels match the wildcard. Example: `www.example.*` matches `www.example.com` and `www.example.co`.
- Comment lines must start with the pound (`#`) character.
- Comments are removed during upload. Files you download contain only the entries, not the comments.

Feed requirements

- When configuring a feed, specify its location using a URL; the URL cannot be Punycode-encoded.
- For feed update intervals of 30 minutes or less, you must provide an MD5 URL to prevent frequent downloads of unchanged feeds. If your feed server does not provide an MD5 URL, use a download interval of at least 30 minutes.
- If you use an MD5 checksum, store it in a simple text file. The file should contain only the checksum. Do not add comments to the file.

List URL syntax and matching criteria for URL lists and feeds

Security Intelligence URL lists and feeds, including custom lists and feeds and entries in the global Block list and Do Not Block list, can include hostnames, URLs, slashes for exact matches, wildcards, and IP addresses, each with specific matching behavior.

- Hostnames

  For example, **`www.example.com`**.

- URLs
  - **`example.com`** matches **`example.com`** and all subdomains, including:
    - **`www.example.com`**
    - **`eu.example.com`**
    - **`example.com/abc`**
    - **`www.example.com/def`**
  - Does not match:
    - **`example.co.uk`**
    - **`examplexyz.com`**
    - **`example.com.malicious-site.com`**
  - When a feed or list includes a single entry, every URL that ends with those domains is identified and blocked
  - Example feed:

    If you add `www.netflix.com, www.amazon.*, org, edu, www.hulu.*` to the feed, these are blocked:
    - `http://www.amazon.in`
    - `http://www.rajiv.org/`
    - `http://www.edu.edu/`
    - `http://www.edu.org/`
    - `http://org.org/`

- `http://edu.edu`

- You can also include an entire URL path, such as **`https://www.cisco.com/c/en/us/products/security/firewalls/index.html`**

- Custom feeds with embedded credentials:

  You can create a custom URL, Network, and DNS feeds, including the username and password inside the URL (for example: **`https://admin:password@server.domain.com/list.txt`**).



Note If your password contains special characters (such as a colon (:) or an at sign (@)), transmission would fail. Ensure that your passwords do not contain these special characters, or use encoded passwords.

- A slash at the end of a URL to specify an exact match

  `example.com/` matches only **`example.com`**; it does not match **`www.example.com`** or any other URL.

- A wildcard (*) to represent any domain in a URL

  - An asterisk can represent a complete domain string separated by dots, but not a partial domain string, and not any part of the URL following the first slash.

    Valid examples:

    - **`*.example.com`**
    - **`www.*.com`**
    - **`example.*`**

      (This will match **`example.com`** and **`example.org`** and **`example.de`**, for example, but not **`example.co.uk`**)

    - **`*.example.*`**
    - **`example.*/`**

    Invalid examples:

    - **`example*.com`**
    - **`example.com/*`**

- IP addresses (IPv4)

  - For IPv6 addresses, or to use ranges or CIDR notation, use the Security Intelligence Network object.
  - You can include one or more wildcards representing an octet, for example 10.10.10.* or 10.10.*.*.

See also Custom Security Intelligence lists, on page 99.

Custom security intelligence feeds

A custom security intelligence feed is a threat intelligence source that

- enables augmentation of system-provided intelligence feeds with regularly updated reputable block lists and do not block lists from the internet,
- supports configuration of internal feeds for updating multiple Secure Firewall Management Center appliances using a single source list, and
- allows use of MD5 checksums to determine if a feed requires downloading, reducing unnecessary updates for large or internal feeds.

Security intelligence feed management

Custom or third-party Security Intelligence feeds allow you to augment the system-provided intelligence feeds with other regularly updated reputable block lists and do not block lists on the internet. You can also set up an internal feed, which is useful if you want to update multiple Secure Firewall Management Center appliances in your deployment using one source list.

You can configure the system to use an MD5 checksum to determine whether to download an updated feed. If the checksum has not changed since the last download, the system does not need to re-download it. Using MD5 checksums is especially helpful for internal feeds that are large.

Note You cannot add address blocks to block or do not block lists using a `/0` netmask in a Security Intelligence feed. If you want to monitor or block all traffic targeted by a policy, use an access control rule with the **Monitor** or **Block** rule action and specify`any` for the **Source Networks** and **Destination Networks**.

Note The system does **not** perform peer SSL certificate verification when downloading custom feeds, nor does the system support the use of certificate bundles or self-signed certificates to verify the remote peer.

If you want strict control over when the system updates a feed from the internet, you can disable automatic updates for that feed. However, automatic updates ensure that the data is always current and relevant.

When you manually update Security Intelligence feeds, the system updates all feeds, including the system-provided intelligence feeds.

See complete requirements at List custom lists and feeds requirements, on page 94.

Create custom security intelligence feeds

Configure security intelligence feeds to enhance your device's ability to detect and respond to threats by importing external or custom threat data sources.

You must have the IPS license (for Firewall Threat Defense devices) or the Protection license (all other device types).

Use this task when you need to add new threat intelligence feeds for IP, DNS, or URL data to your security device.

Procedure

Step 1 Choose **Objects** > **Object Management** > **Security Intelligence** node, then choose a feed type you want to add.

Step 2 Click the option appropriate to the feed type you chose:

- **Add Network Lists and Feeds** (for IP addresses)
- **Add DNS Lists and Feeds**
- **Add URL Lists and Feeds**

Step 3 Enter a **Name** for the feed.

Step 4 Choose **Feed** from the **Type** drop-down list.

Step 5 Enter a **Feed URL**.

Step 6 Enter an **MD5 URL**.

- The MD5 URL is used to check if the feed content has changed since the last update and is required for update intervals shorter than 30 minutes.
- If your feed provider does not supply an MD5 URL, set the update interval to at least 30 minutes.

Step 7 Choose an **Update Frequency**.

Step 8 Click **Save**.
Unless you disabled feed updates, the system downloads and verifies the feed.

Update custom security intelligence feeds manually

Manually updating Security Intelligence feeds allows you to ensure that your deployment uses the most current threat intelligence data for filtering network traffic.

You must have the IPS license (for Firewall Threat Defense devices) or the Protection license (all other device types).

Before you begin

At least one device must already be added to the management center.

Procedure

Step 1 Choose **Objects** > **Object Management** > **Security Intelligence**.

Step 2 Click **Update Feeds**, then confirm.

Step 3 Click **OK**.

After the Secure Firewall Management Center downloads and verifies the feed updates, it communicates any changes to its managed devices. Your deployment begins filtering traffic using the updated feeds.

Custom Security Intelligence lists

A custom Security Intelligence list is a static list that

- contains IP addresses, address blocks, URLs, or domain names that you manually upload to the system,
- enables you to augment and fine-tune feeds or global lists for the managed devices of a single Firewall Management Center, and
- is useful when you need to override or supplement existing feed behavior without removing the entire feed from a policy.

Formatting requirements for custom Security Intelligence lists

- Netmasks for address blocks must be integers from `0` to `32` for IPv4 or from `0` to `128` for IPv6.
- Encode Unicode in domain names using Punycode format. The encoding is not case sensitive.
- Characters in domain names are not case sensitive.
- Unicode in URLs must be percent-encoded.
- Characters in URL subdirectories are case sensitive.
- List entries that start with the number sign (`#`) are treated as comments.
- For more information about formatting requirements, see the List custom lists and feeds requirements, on page 94.

Matching requirements for custom Security Intelligence lists

- If you add a higher-level domain to a URL or DNS list, any sub-level domains also match. For example, if you add `example.com` to a DNS list, `www.example.com` and `test.example.com` also match.
- DNS lookups (forward or reverse) are not performed on DNS or URL list entries. For example, if you add `http://198.51.100.2` to a URL list, the list matches only that exact address and not any domain it resolves to.



Note You cannot add address blocks to a Block or Do Not Block list using a `/0` netmask in a Security Intelligence list. To monitor or block all traffic targeted by a policy, use an access control rule with the **Monitor** or **Block** action, and set the **Source Networks** and **Destination Networks** to `any`.

Examples for customizing Security Intelligence lists

- If a global feed blocks access to a vital resource but is otherwise useful, you can upload a custom list to allow access to that resource without disabling the feed entirely.
- You can supplement an existing feed by adding URLs or domains relevant to your organization.

Upload new custom security intelligence lists to the Secure Firewall Management Center

To modify a Security Intelligence list, you must make your changes to the source file and upload a new copy. You cannot modify the file's contents using the web interface. If you do not have access to the source file, download a copy from the system.

Procedure

Step 1 Choose **Objects** > **Object Management** > **Security Intelligence** node, then choose a list type.

Step 2 Click the option appropriate to the list you chose above:

- **Add Network Lists and Feeds** (for IP addresses)
- **Add DNS Lists and Feeds**
- **Add URL Lists and Feeds**

Step 3 Enter a **Name**.

Step 4 From the **Type** drop-down list, choose **List**.

Step 5 Click **Browse** to browse to the list `.txt` file, then click **Upload**.

Step 6 Click **Save**.

What to do next

You do not need to redeploy these changes to take effect. If you want to delete an entry from the list, see Delete entries from global security intelligence lists, on page 92.

Update custom security intelligence lists

Modify, add, or remove entries from your security intelligence lists in Secure Firewall Management Center. Update your lists when new threats emerge or when you need to refine your firewall's response to known risks. Perform this task as part of routine security maintenance or to update your firewall based on new threat intelligence.

Procedure

Step 1 Choose **Objects** > **Object Management** > **Security Intelligence** node, then choose a list type.

Step 2 Next to the list you want to update, click **Edit** ( ).

If **View** ( ) appears instead, the configuration belongs to an ancestor domain, or you do not have permission to modify the configuration.

Step 3 To edit the list, click **Download** and save the list as a text file using your browser's prompts.

Step 4 Change the list as needed.

Step 5 On the Security Intelligence pop-up window, click **Browse** to browse to the modified list, then click **Upload**.

Step 6 Click **Save**.

What to do next

These changes take effect without redeployment. If you want to delete an entry from the list, see Delete entries from global security intelligence lists, on page 92.

Sinkhole objects

A sinkhole object is a network security object that

- represents either a DNS server that gives non-routable addresses for all domain names within the sinkhole
- can represent an IP address that does not resolve to a server, and
- requires assignment of both an IPv4 address and an IPv6 address.

Sinkhole object usage in DNS policy rules

You can reference the sinkhole object within a DNS policy rule to redirect matching traffic to the sinkhole.

Create a sinkhole object

Create a sinkhole object to redirect or block suspicious traffic for enhanced threat detection and analysis.

Use this task when you need to configure a sinkhole object for traffic redirection or blocking to maintain security.

Before you begin

Ensure you have the IPS license (for Firewall Threat Defense devices) or the Protection license (all other device types).

Procedure

Step 1 Choose **Objects** > **Object Management** > **Sinkhole**.

Step 2 Click **Add Sinkhole**.

Step 3 Enter a **Name**.

Step 4 Enter the **IPv4 Address** and **IPv6 Address** of your sinkhole.

Step 5 You have these options:

- To redirect traffic to a sinkhole server, choose **Log Connections to Sinkhole**.
- To redirect traffic to a non-resolving IP address, choose **Block and Log Connections to Sinkhole**.

Step 6 (Optional) From the **Type** drop-down list, assign an Indication of Compromise (IoC) type for the sinkhole object, if needed.

Step 7 Click **Save**.

Configure an SLA monitor

Configure an SLA monitor to define a connectivity policy to a monitored address and track the availability of a route. This helps ensure that the route is periodically checked and replaced with a backup route if it becomes unavailable.

An Internet Protocol Service Level Agreement (SLA) monitor defines a connectivity policy to a monitored address and tracks the availability of a route. The monitor periodically checks the route by sending ICMP echo requests. If requests time out, the route is removed from the routing table and replaced with a backup route. SLA monitoring jobs start immediately after deployment and continue to run unless you remove the SLA monitor from the device configuration. The SLA Monitor Object is used in the Route Tracking field of an IPv4 Static Route Policy.



Note IPv6 routes do not support SLA monitors via route tracking.

Procedure

Step 1 Select **Objects** > **Object Management** > **SLA Monitor**.

Step 2 Click **Add SLA Monitor** and enter these values:

Fields	Details
Name	Enter a name for the object.
Description	Enter a description for the object.
Frequency	Enter the frequency of ICMP echo request transmissions, in seconds. Valid values range from 1 to 604800 seconds (7 days). The default is 60 seconds. **Note** The frequency cannot be less than the timeout value; you must convert frequency to milliseconds to compare the values.
SLA Monitor ID	Enter the ID number of the SLA operation. Values range from 1 to 2147483647. You can create a maximum of 2000 SLA operations on a device. Each ID number must be unique to the policy and the device configuration.
Threshold	Enter the amount of time that must pass after an ICMP echo request before a rising threshold is declared, in milliseconds. Valid values range from 0 to 2147483647 milliseconds. The default is 5000 milliseconds. The threshold value identifies events that exceed the defined value, which you can use to evaluate the proper timeout value. However, the threshold does not serve as a direct indicator of the reachability of the monitored address. **Note** The threshold value should not exceed the timeout value.

Fields	Details
Timeout	Enter the amount of time that the SLA operation waits for a response to the ICMP echo requests, in milliseconds. Values range from 0 to 604800000 milliseconds (7 days). The default is 5000 milliseconds. If a response is not received from the monitored address within the amount of time defined in this field, the static route is removed from the routing table and replaced by the backup route. **Note** The timeout value cannot exceed the frequency value (adjust the frequency value to milliseconds to compare the numbers).
Data Size	Enter the size of the ICMP request packet payload, in bytes. Values range from 0 to 16384 bytes. The default is 28 bytes, which creates a total ICMP packet of 64 bytes. Do not set this value higher than the maximum allowed by the protocol or the Path Maximum Transmission Unit (PMTU). For purposes of reachability, you might need to increase the default data size to detect PMTU changes between the source and the target. A low PMTU can affect session performance and, if detected, might indicate that the secondary path should be used.
ToS	Enter a value for type of service (ToS) defined in the IP header of the ICMP request packet. Values range from 0 to 255. The default is 0. This field contains information such as delay, precedence, reliability, and so on. It can be used by other devices on the network for policy routing and features such as committed access rate.
Number of Packets	Enter the number of packets that are sent. Values range from 1 to 100. The default is 1 packet. **Note** Increase the default number of packets if you are concerned that packet loss might falsely cause the Secure Firewall Threat Defense device to believe that the monitored address cannot be reached.
Monitored Address	Enter the IP address that is being monitored for availability by the SLA operation.

Step 3 The **Available Zones** list displays both zones and interface groups. In the **Zones/Interfaces** list, add the zones or interface groups that contain the interfaces through which the device communicates with the management station. To specify a single interface, you need to create a zone or the interface groups for the interface; see Create security zone and interface group objects. The host will be configured on a device only if the device includes the selected interfaces or zones.

Step 4 Click **Save**.

After completing these steps, the SLA monitor is configured and begins tracking the availability of the specified route. If the monitored address becomes unreachable, the route is removed from the routing table and replaced with a backup route.

Time ranges

A time range is a configuration object that

- defines specific time periods,
- enables use of these periods to determine when rules apply, and
- supports integration with time-based ACLs.

Time range objects allow you to specify periods for rule enforcement.

Note Time-based ACLs is supported in Snort 3 also from Firewall Management Center 7.0 onwards.

Create a time range object

Establish a time range object so policies apply only during specified periods, enabling precise control of network traffic.

If you want a policy to apply only during a specified time range, create a time range object and specify that object in the policy. This object works on Firewall Threat Defense devices only.

You can specify time range objects only in certain policy types, which are listed in this topic.

Note The timezone represents the device's local time and is used only for applying the time ranges in rules in the policies that support the time ranges. The timezone does not change the configured time of the device. To verify the configuration, in the Firewall Threat Defense CLI, use the **show time-range timezone** and **show time** commands (see the *Cisco Secure Firewall Threat Defense Command Reference* guide). In addition, the timezone of a chassis overrides the management center timezone.

Before you begin

Time ranges operate based on the time zone associated with the device that processes the traffic. By default, this is UTC. To change the time zone associated with a device, go to **Devices** > **Platform Settings**.

Procedure

Step 1 Choose **Objects** > **Object Management** > **Time Range** from the list of object types.

Step 2 Click **Add Time Range**.

Step 3 Enter the object name and specify the desired time periods.

- If you see a red error box around the object name you have entered, mouse over the **Name** field to see naming restrictions.
- All times are in UTC, unless you specify a time zone for the device in **Devices** > **Platform Settings**.

- Enter times using a 24-hour clock; for example, enter 13:30 for 1:30 PM.
- To specify a single continuous range, such as typical weekend hours (Fridays at 5pm through Mondays at 8am, including evenings and nights), choose Range Type **Range**.
- To specify part of multiple days, such as Monday through Friday from 8am to 5pm (excluding evenings, nights, and early mornings every day), choose Range Type **Daily Interval**.
- You may specify up to 28 periods per object.
- For multiple noncontiguous times of day or different hours for different days, create multiple recurring intervals. For example, to apply a policy at all times other than standard working hours, create a single time range object with the following two recurring intervals:
  - A Daily Interval for Monday through Friday from 5pm through 8am, and
  - A Range recurring interval for Friday at 5pm through Monday at 8am.

Step 4 Click **Save**.

What to do next

Configure time ranges in any of these:

- Access control rules
- Prefilter rules
- Tunnel rules
- VPN group policy

In a VPN group policy object, specify the time range object using the **Access Hours** field. For details, see Configure a group policy object, on page 130 and Advanced group policy option fields and values, on page 136.

Time zones

A time zone is a configuration object that

- specifies the local time zone for a managed device
- is used only for applying time ranges in rules in policies that support time ranges, such as access control, prefilter, and VPN group policies), and
- defaults to UTC if not assigned to a device.

Device support and usage of time zones

- Time zone objects are supported only for Firewall Threat Defense devices.
- Time-based ACLs is supported in Snort 3 also from Firewall Management Center 7.0 onwards.

Tunnel zones

A tunnel zone is a network object that

- represents certain types of plaintext, passthrough tunnels that you explicitly tag for special analysis,
- is not an interface object, and
- can be used as an interface constraint in some configurations.

For detailed information, see Using tunnel zones to apply access control at the tunnel level.

URLs

A URL object is a configuration element that

- defines a single URL or IP address for use within access control policies, event searches, and Security Intelligence configurations,
- can be used in multiple locations throughout the system's web interface to specify what traffic is filtered or monitored, and
- provides distinct matching behavior compared to URL group objects, which define multiple URLs or addresses.

URL object configuration guidelines



Important For best practices for using this and similar options in Security Intelligence configurations and for URL rules in access control and QoS policies, see Manual URL filtering options.

When creating URL objects, consider the following guidelines:

- If the URL does not include a path (no / characters), the match is based only on the server's hostname. If a path is included (one or more / characters), the entire URL string is used for a substring match.
- A URL is considered a match if:
  - the string appears at the beginning of the URL,
  - the string follows a dot,
  - the string contains a dot at the beginning,
  - or the string follows the :// characters.
- The system ignores the encryption protocol (HTTP or HTTPS). Blocking a website applies to both unless an application condition targets a specific protocol. Specify URLs without the protocol (use example.com, not http://example.com).
- To match HTTPS traffic in access control rules, use the subject common name from the public key certificate. Do not include subdomain information. The subject common name in certificates may differ

from the web site's domain (for example, youtube.com uses *.google.com). Use SSL Decryption policy to decrypt HTTPS traffic for consistent URL filtering results.

- URL objects will not match HTTPS traffic if the browser resumes a TLS session, as certificate information may not be available, which can lead to inconsistent results.
- When using URL objects as aliases in remote access VPN policies, ensure the URL is 128 characters or less.
- Do not use manual URL filtering to block or allow individual web pages or parts of sites (that is, URL strings with / characters), as servers can be reorganized and pages moved to new paths.

URL object matching examples

For example, ign.com matches ign.com or www.ign.com, but not versign.com. Use example.com rather than http://example.com when specifying a URL object.

Create a URL object

Use this task when you need to define a new URL object in the Secure Firewall Management Center for use in access control or other security policies.

Procedure

Step 1 Choose **Objects** > **Object Management** > **URL** from the list of object types.

Step 2 Choose **Add Object** from the **Add URL** drop-down list.

Step 3 Enter a **Name**.

Step 4 (Optional) Enter a **Description**.

Step 5 Enter the **URL** or IP address.

Step 6 (Optional) Manage overrides for the object:

- If you want to allow overrides for this object, check the **Allow Overrides** check box; see Allow object overrides, on page 14.
- If you want to add override values to this object, expand the Override section and click **Add**; see Add object overrides, on page 14.

Step 7 Click **Save**.

What to do next

- If an active policy references your object, deploy configuration changes; see Deploy configuration changes.

Variable sets

A variable set is a configuration group that

- collects and manages variables used in intrusion rules and policies
- allows customization of predefined and user-defined variables, and
- enables linking of variable sets to intrusion policies for flexible network security management.

Variable set management and usage

Variables represent values commonly used in intrusion rules to identify source and destination IP addresses and PORTS. You can also use variables in intrusion policies to represent IP addresses in rule suppressions, adaptive profile updates, and dynamic rule states.

Tip Preprocessor rules can trigger events regardless of the hosts defined by network variables used in intrusion rules.

You use variable sets to manage, customize, and group your variables. You can use the default variable set provided by the system or create your own custom sets. Within any set you can modify predefined default variables and add and modify user-defined variables.

Most of the shared object rules and standard text rules that the system provides use predefined default variables to define networks and port numbers. For example, the majority of the rules use the variable `$HOME_NET` to specify the protected network and the variable `$EXTERNAL_NET` to specify the unprotected (or outside) network. In addition, specialized rules often use other predefined variables. For example, rules that detect exploits against web SERVERS use the `$HTTP_SERVERS` and `$HTTP_PORTS` variables.

Rules are more effective when variables more accurately reflect your network environment. At a minimum, you should modify default variables in the default set. By ensuring that a variable such as `$HOME_NET` correctly defines your network and `$HTTP_SERVERS` includes all web SERVERS on your network, processing is optimized and all relevant systems are monitored for suspicious activity.

To use your variables, you link variable sets to intrusion policies associated with access control rules or with the default action of an access control policy. By default, the default variable set is linked to all intrusion policies used by access control policies.

Adding a variable to any set adds it to all sets; that is, each variable set is a collection of all variables currently configured on your system. Within any variable set, you can add user-defined variables and customize the value of any variable.

Initially, the system provides a single, default variable set comprised of predefined default values. Each variable in the default set is initially set to its default value, which for a predefined variable is the value set by the Talos Intelligence Group and provided in rule updates.

Although you can leave predefined default variables configured to their default values, Cisco recommends that you modify a subset of predefined variables.

You could work with variables only in the default set, but in many cases you can benefit most by adding one or more custom sets, configuring different variable values in different sets, and perhaps even adding new variables.

When using multiple sets, it is important to remember that the *current value* of any variable in the default set determines the *default value* of the variable in all other sets.

When you select **Variable Sets** on the Object Manager page, the object manager lists the default variable set and any custom sets you created.

On a freshly installed system, the default variable set is comprised only of the default variables predefined by Cisco.

Each variable set includes the default variables provided by the system and all custom variables you have added from any variable set. Note that you can edit the default set, but you cannot rename or delete the default set.



Caution Importing an access control or an intrusion policy overwrites existing default variables in the default variable set with the imported default variables. If your existing default variable set contains a custom variable not present in the imported default variable set, the unique variable is preserved.

Related Topics

Manage variables, on page 121
Manage variable sets, on page 120

Variable sets in intrusion policies

A variable set is a configuration group that

- links to intrusion policies used in an access control policy,
- provides variable values for enabled intrusion rules, and
- affects deployment status when modified.

Deployment status for variable sets in intrusion policies

By default, the system links the default variable set to all intrusion policies used in an access control policy. When you deploy an access control policy that uses an intrusion policy, intrusion rules that you have enabled in the intrusion policy use the variable values in the linked variable set.

When you modify a custom variable set used by an intrusion policy in an access control policy, the system reflects the status for that policy as out-of-date on the Access Control Policy page. You must re-deploy the access control policy to implement changes in your variable set. When you modify the default set, the system reflects the status of all access control policies that use intrusion policies as out-of-date, and you must re-deploy all access control policies to implement your changes.

Variables

A variable is a configuration element that

- can be provided by the system or created by users
- is categorized as default, customized, or advanced, and
- enables flexible management of network addresses and ports in rule creation.

Variable categories

Variables are grouped into three categories: default, customized, and advanced.

- **Default variables** are provided by the system. You cannot rename or delete a default variable, and you cannot change its default value. However, you can create a customized version of a default variable.
- **Customized variables** are variables you create. These variables can include:
  - *customized default variables*

    When you edit the value for a default variable, the system moves the variable from the Default Variables area to the Customized Variables area. Variable values in the default set determine the default values of variables in custom sets. If you customize a default variable in the default set, the system changes the default value of the variable in all other sets.
  - *user-defined variables*

    You can add and delete your own variables, customize their values within different variable sets, and reset customized variables to their default values. When you reset a user-defined variable, it remains in the Customized Variables area.

    User-defined variables can be one of these types:
    - *network* variables specify the IP addresses of hosts in your network traffic.
    - *port* variables specify TCP or UDP ports in network traffic, including the value `any` for either type.
- **Advanced variables** are provided by the system under specific conditions. These variables have a very limited deployment.

Variable usage examples

For example, if you create custom standard text rules, you might also want to add your own user-defined variables to more accurately reflect your traffic or as shortcuts to simplify the rule creation process. Alternatively, if you create a rule that you want to inspect traffic in the "demilitarized zone" (or DMZ) only, you can create a variable named `$DMZ` whose value lists the server IP addresses that are exposed. You can then use the `$DMZ` variable in any rule written for this zone.

List predefined default variables

By default, the system provides a single default variable set, which is comprised of predefined default variables. The Talos Intelligence Group uses rule updates to provide new and updated intrusion rules and other intrusion policy elements, including default variables. Because many intrusion rules from the system use predefined default variables, set appropriate values for these variables. You can modify the values for these default variables in any or all variable sets, depending on how you use variable sets to identify traffic on your network.



Caution Importing an access control or an intrusion policy overwrites existing default variables in the default variable set with the imported default variables. If your existing default variable set contains a custom variable not present in the imported default variable set, the unique variable is preserved.

This table describes the variables provided by the system and indicates which variables you typically would modify. For assistance determining how to tailor variables to your network, contact Professional Services or Support.

Table 4: System-Provided variables

Variable Name	Description	Modify?
`$AIM_SERVERS`	Defines known AOL Instant Messenger (AIM) SERVERS, and is used in chat-based rules and rules that look for AIM exploits.	Not required.
`$DNS_SERVERS`	Defines Domain Name Service (DNS) SERVERS. If you create a rule that affects DNS SERVERS specifically, you can use the `$DNS_SERVERS` variable as a destination or source IP address.	Not required in curre
`$EXTERNAL_NET`	Defines the network that the system views as the unprotected network, and is used in many rules to define the EXTERNAL network.	Yes, you should adeq `$HOME_NET` and then e `$HOME_NET` as the val `$EXTERNAL_NET`.
`$FILE_DATA_PORTS`	Defines non-encrypted ports used in intrusion rules that detect files in a network stream.	Not required.
`$FTP_PORTS`	Defines the ports of FTP SERVERS on your network, and is used for FTP server exploit rules.	Yes, if your FTP SER ports other than the d (you can view the de the web interface).
`$GTP_PORTS`	Defines the data channel ports where the packet decoder extracts the payload inside a GTP (General Packet Radio Service [GPRS] Tunneling Protocol) PDU.	Not required.
`$HOME_NET`	Defines the network that the associated intrusion policy monitors, and is used in many rules to define the internal network.	Yes, to include the IP your internal network
`$HTTP_PORTS`	Defines the ports of web SERVERS on your network, and is used for web server exploit rules.	Yes, if your web SER ports other than the d (you can view the de the web interface).
`$HTTP_SERVERS`	Defines the web SERVERS on your network. Used in web server exploit rules.	Yes, if you run HTTP
`$ORACLE_PORTS`	Defines Oracle database server ports on your network, and is used in rules that scan for attacks on Oracle databases.	Yes, if you run Oracl
`$SHELLCODE_PORTS`	Defines the ports you want the system to scan for shell code exploits, and is used in rules that detect exploits that use shell code.	Not required.
`$SIP_PORTS`	Defines the ports of SIP SERVERS on your network, and is used for SIP exploit rules.	Not required.

Variable Name	Description	Modify?
$SIP_SERVERS	Defines SIP SERVERS on your network, and is used in rules that address SIP-targeted exploits.	Yes, if you run SIP SERV should adequately define $HOME_NET and then inclu $HOME_NET as the value f $SIP_SERVERS.
$SMTP_SERVERS	Defines SMTP SERVERS on your network, and is used in rules that address exploits that target mail SERVERS.	Yes, if you run SMTP SE
$SNMP_SERVERS	Defines SNMP SERVERS on your network, and is used in rules that scan for attacks on SNMP SERVERS.	Yes, if you run SNMP SE
$SNORT_BPF	Identifies a legacy advanced variable that appears only when it existed on your system in a software release before Version 5.3.0 that you subsequently upgraded to Version 5.3.0 or greater.	No, you can only view or variable. You cannot edit recover it after deleting i
$SQL_SERVERS	Defines database SERVERS on your network, and is used in rules that address database-targeted exploits.	Yes, if you run SQL SER
$SSH_PORTS	Defines the ports of SSH SERVERS on your network, and is used for SSH server exploit rules.	Yes, if your SSH SERVE ports other than the defau (you can view the defaul the web interface).
$SSH_SERVERS	Defines SSH SERVERS on your network, and is used in rules that address SSH-targeted exploits.	Yes, if you run SSH SER you should adequately de $HOME_NET and then inclu $HOME_NET as the value f $SSH_SERVERS.
$TELNET_SERVERS	Defines known Telnet SERVERS on your network, and is used in rules that address Telnet server-targeted exploits.	Yes, if you run Telnet SE
$USER_CONF	Provides a general tool that allows you to configure one or more features not otherwise available via the web interface. Conflicting or duplicate $USER_CONF configurations will halt the system.	No, only as instructed in description or with the gu Support.

Network variables

A network variable is a configuration element that

- represents IP addresses for use in intrusion rules, intrusion policy rule suppressions, dynamic rule states, and adaptive profile updates
- differs from network objects and network object groups by being specific to intrusion policies and intrusion rules, and
- enables specification of host IP addresses in various configurations within an intrusion policy.

Network variable usage and configuration

Network variables can be used to specify the IP addresses of hosts on the network in the following configurations:

- Intrusion rules: The **Source IPs** and **Destination IPs** fields restrict packet inspection to traffic originating from or destined to specific IP addresses.
- Suppressions: The **Network** field in source or destination intrusion rule suppressions suppresses intrusion event notifications when a specific IP address or range triggers an intrusion rule or preprocessor.
- Dynamic rule states: The **Network** field in dynamic rule states identifies when too many matches for an intrusion rule or preprocessor occur within a given time period.
- adaptive profile updates: The**Networks** field identifies hosts where packet fragment and TCP stream reassembly is improved in passive deployments.

When variables are used in these fields, the variable set linked to an intrusion policy determines the variable values applied to network traffic handled by the related access control policy.

A variable can include any combination of the following network configurations:

- network variables, network objects, and network object groups selected from the available networks list,
- individual network objects added from the New or Edit Variable page, which can then be used in variables or other configurations,
- literal single IP addresses or address blocks; multiple literal IP addresses and address blocks may be included individually, supporting both IPv4 and IPv6 addresses.

  When specifying IPv6 addresses, any addressing convention defined in RFC 4291 may be used.

The default value for included networks in any variable is `any`, which indicates any IPv4 or IPv6 address. The default value for excluded networks is `none`, meaning no network. If you specify the address `::`, it means any IPv6 address when included, or no IPv6 addresses when excluded.

Adding networks to the excluded list negates those addresses and blocks; all addresses except the excluded ones can be matched.

For example, excluding `192.168.1.1` specifies any IP address other than 192.168.1.1. Excluding `2001:db8:ca2e::fa4c`specifies any IP address except 2001:db8:ca2e::fa4c.

Exclusions can be made with any combination of literal or available networks. For instance, excluding both `192.168.1.1` and `192.168.1.5` matches any IP address other than 192.168.1.1 or 192.168.1.5.

Keep these points in mind when you add or edit network variables:

- The value `any` cannot be logically excluded, as it would mean no address; for example, it is not possible to add a variable with the value `any` to the exclusion list.
- Network variables identify traffic for specific intrusion rule and intrusion policy features; preprocessor rules can trigger events regardless of the hosts defined by network variables.
- Excluded values must resolve to a subset of included values; for example, including the address block 192.168.5.0/24 and excluding 192.168.6.0/24 is not valid.

Port variables

A port variable is a network configuration element that

- represents TCP and UDP ports you can use in the **Source Port** and **Destination Port** header fields in intrusion rules,
- is specific to intrusion rules and differs from port objects and port object groups, and
- can be used to restrict packet inspection to packets originating from or destined to specific TCP or UDP ports.

Port variable configuration details

Port variables are linked to intrusion policies associated with access control rules or policies, determining the values for these variables in the network traffic where you deploy the access control policy.

When configuring a port variable, you can include:

- any combination of port variables and port objects that you select from the list of available ports (excluding port object groups, which cannot be added to variables),
- individual port objects that you add while creating or editing a variable, which can also be reused for other variables,
- and single literal port values as well as port ranges (ranges must use a dash, e.g., 80-90; colon-separated ranges are supported for backward compatibility but not for new variables).

  You can list multiple literal port values and ranges by adding each individually in any combination.

Port variable configuration has these important considerations:

- The default value for included ports in any variable is `any`, indicating all ports or port ranges. The default for excluded ports is `none`, indicating no ports.



Tip To create a variable with the value `any`, name and save the variable without adding a specific value.

- You cannot exclude the value `any` from a variable, as that would result in no ports being matched.
- Adding ports to an excluded list prevents matching on those specified ports or port ranges. The system matches any port except those excluded.
- Excluded values must be a subset of the included values. For example, if you include the port range 10-50, you cannot exclude port 60.

Advanced variables

An advanced variable is a configuration mechanism that

- enables you configure of features not available through the web interface
- includes the USER_CONF variable as your primary advanced variable, and
- allows you to enter custom instructions or lines up to the maximum character length.

User_CONF variable details

The USER_CONF variable provides a general tool that allows configuration of one or more features not otherwise available via the web interface.

Caution Do **not** use the advanced variable USER_CONF to configure an intrusion policy feature unless you are instructed to do so in the feature description or by Support. Conflicting or duplicate configurations will halt the system.

When editing USER_CONF, you can type up to 4096 characters on a single line. The line wraps automatically. You can include any number of valid instructions or lines until you reach either the 8192-character maximum length for a variable or a physical limit, such as disk space. Use the backslash (\) character as a line continuation after any complete argument in a command directive.

Resetting USER_CONF empties it.

Variable resets

A variable reset is a configuration action that

- returns a variable to its default value on the variable set new or edit variables page,
- applies different reset behaviors depending on whether the variable is in a default or custom set, and
- affects the value used in any intrusion policy where the variable set is linked.

Variable reset values

You can reset a variable to its default value on the variable set new or edit variables page. This table summarizes the basic principles of resetting variables.

Table 5: Variable reset values

Resetting this variable type...	In this set type...	Resets it to...
default	default	the rule update value
user-defined	default	`any`
default or user-defined	custom	the current default set value (modified or unmodified)

When you reset a variable in a custom set, you return it to the current value for that variable in the default set.

If you reset or change a variable in the default set, you update the default value in all custom sets. If the reset icon is unavailable, you cannot reset the variable because it has no customized value in the set. If you have not customized the variable in a custom set, changing it in the default set also updates any intrusion policy where you have linked the variable set.

Note Before you modify a variable in the default set, check how this change affects any intrusion policy that uses the variable in a linked custom set, especially if you have not customized the variable value in the custom set.

You can hover your pointer over the **Reset** icon in a variable set to see the reset value. If the customized value and the reset value match, you are in one of these situations:

- you are in the custom or default set where you added the variable with the value `any`.
- you are in the custom set where you added the variable with an explicit value and elected to use the configured value as the default value.

Variable sets

A variable set is a configuration grouping that

- allows variables to be managed collectively across multiple sets
- enables the addition of variables to all sets simultaneously, and
- provides options for using configured or default values when adding variables.

Variable set behavior

When you add a variable to a variable set, it is also added to all other sets. If you add a variable from a custom set, decide if you want to use the configured value as the customized value in the default set.

- **If you use the configured value** (for example, 192.168.0.0/16), the variable is added to the default set using the configured value as a customized value with a default value of `any`. The current value in the default set determines the default value in other sets. Therefore, the initial default value in other custom sets is the configured value (which in this example is 192.168.0.0/16).
- **If you do not use the configured value**, the variable is added to the default set using only the default value `any` and, consequently, the initial, default value in other custom sets is `any`.

Examples: Adding user-defined variables to default sets

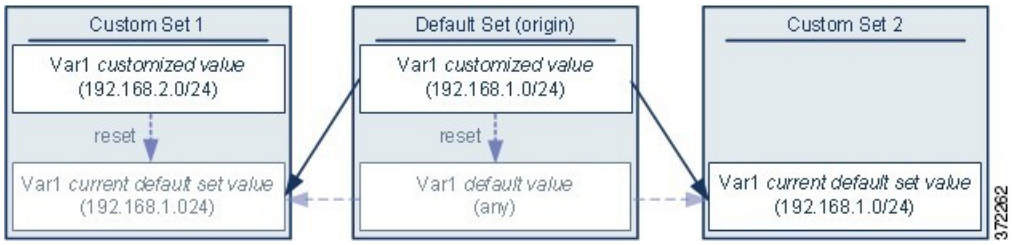
A user-defined variable addition is a configuration action that

- enables adding a variable, such as Var1, to a default set with a specific value (like 192.168.1.0/24),
- allows customization of the variable value in any set, and
- ensures that resetting the variable in the default set resets its default value to `any` across all sets.

Set interactions and variable customization

The diagram below illustrates how adding and customizing user-defined variables affects default and custom sets.

Figure 3: Set interaction diagram



Customizing the value of `Var1` in a custom set overrides the default value. For example, if `Var1` is customized in `Custom Set 1`, it takes on a new value, while in `Custom Set 2`, if `Var1` is not customized, it retains the default value (e.g., `192.168.1.0/24`). Resetting the user-defined variable in the default set updates its default value to `any` in all sets.

Further, when `Var1` in `Custom Set 2` is not updated, customizing or resetting `Var1` in the default set updates its current value in `Custom Set 2`, affecting any intrusion policy linked to the variable set.

Note While the interactions are the same for user-defined variables and default variables, resetting a default variable in the default set resets it to the value configured by Cisco in the current rule update, not `any`.

Customizing user-defined variable values

In Custom Set 1, the customized value `192.168.2.0/24` of `Var1` overrides the default value, while in Custom Set 2, `Var1` retains the default value `192.168.1.0/24` unless customized.

Resetting default variables in default sets

Although not shown in the example, note that interactions between sets are the same for user-defined variables and default variables except that resetting a default variable in the default set resets it to the value configured by Cisco in the current rule update.

Examples: User-defined variables in custom sets

A user-defined variable in a custom set is a configuration element that:

- enables you to add variables to custom sets,
- allows you to choose whether the configured value becomes the default for other sets, and
- affects how variable values and interactions propagate across sets.

Variable set interactions and default value behavior

When you add a user-defined variable to a custom set, you are prompted to decide if the configured value should be used as the default value for other sets. The examples illustrate the outcomes of each choice.

- Choosing to use the configured value as the default copies the value to the default set as a customized value, with a default value of `any`.

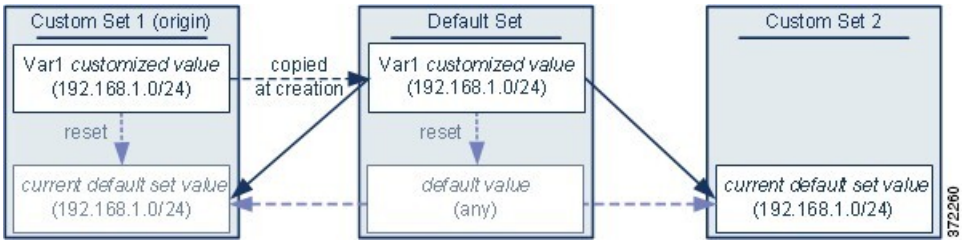
- Choosing not to use the configured value as the default adds the variable to all sets with a default value of `any`, allowing customization in any set.

Note Further customizing or resetting a variable in the default set updates the current, default value in other sets, which can affect any intrusion policy linked to the variable set.

Examples of adding user-defined variables to custom sets

The first example shows adding `Var1` with the value `192.168.1.0/24` to Custom Set 1 and electing **to use** the configured value as the default for other sets.

Figure 4: Variable set interaction when using the configured value as default

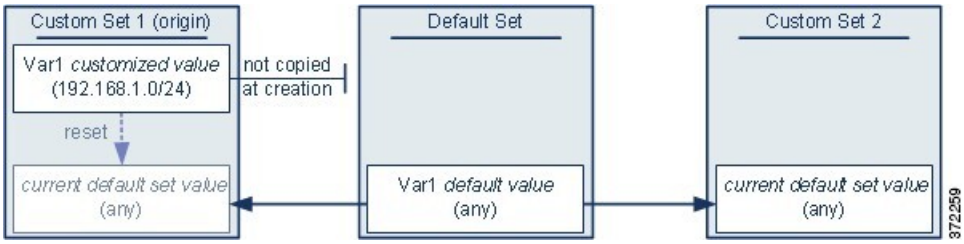


Except for the origin of `Var1` from Custom Set 1, this example is identical to adding `Var1` to the default set. Adding the customized value `192.168.1.0/24` for `Var1` to Custom Set 1 copies the value to the default set as a customized value with a default value of `any`. Thereafter, `Var1` values and interactions are the same as if you had added `Var1` to the default set. Further customizing or resetting `Var1` in the default set updates the current, default value of `Var1` in Custom Set 2, affecting any intrusion policy linked to the variable set.

Adding a variable and not using the configured value as default

The second example shows adding `Var1` with the value `192.168.1.0/24` to `Custom Set 1`, but electing **not to use** the configured value as the default in other sets.

Figure 5: Variable set interaction when not using the configured value as default



This approach adds `Var1` to all sets with a default value of `any`. After adding `Var1`, you can customize its value in any set. An advantage of this approach is that, by not initially customizing `Var1` in the default set, you decrease your risk of customizing the value in the default set and inadvertently changing the current value in a set such as Custom Set 2 where you have not customized `Var1`.

Nesting variables

A nesting variable is a configuration element that

- can include other variables as part of its definition
- does not support circular nesting, and
- does not support nested, negated variables.

Valid nested variables

In this example, SMTP_SERVERS, HTTP_SERVERS, and OTHER_SERVERS are valid nested variables.

Variable	Type	Included Networks	Excluded Networks
SMTP_SERVERS	customized default	10.1.1.1	—
HTTP_SERVERS	customized default	10.1.1.2	—
OTHER_SERVERS	user-defined	10.2.2.0/24	—
HOME_NET	customized default	10.1.1.0/24 OTHER_SERVERS	SMTP_SERVERS HTTP_SERVERS

Invalid nested variable

In this example, HOME_NET is an invalid nested variable because the nesting of HOME_NET is circular; that is, the definition of OTHER_SERVERS includes HOME_NET, so you would be nesting HOME_NET in itself.

Variable	Type	Included Networks	Excluded Networks
SMTP_SERVERS	customized default	10.1.1.1	—
HTTP_SERVERS	customized default	10.1.1.2	—
OTHER_SERVERS	user-defined	10.2.2.0/24 HOME_NET	—
HOME_NET	customized default	10.1.1.0/24 OTHER_SERVERS	SMTP_SERVERS HTTP_SERVERS

Unsupported nested, negated variable

Because nested, negated variables are not supported, do not use the variable NONCORE_NET to represent IP addresses outside your protected networks.

Variable	Type	Included Networks	Excluded Networks
HOME_NET	customized default	10.1.0.0/16 10.2.0.0/16 10.3.0.0/16	—
EXTERNAL_NET	customized default	—	HOME_NET
DMZ_NET	user-defined	10.4.0.0/16	—
NOT_DMZ_NET	user-defined	—	DMZ_NET
NONCORE_NET	user-defined	EXTERNAL_NET NOT_DMZ_NET	—

Alternative to unsupported nested, negated variable

As an alternative to the example above, you could represent IP addresses that are outside of your protected networks by creating the variable NONCORE_NET to represent IP addresses outside your protected networks.

Variable	Type	Included Networks	Excluded Networks
HOME_NET	customized default	10.1.0.0/16 10.2.0.0/16 10.3.0.0/16	—
DMZ_NET	user-defined	10.4.0.0/16	—
NONCORE_NET	user-defined	—	HOME_NET DMZ_NET

Manage variable sets

Manage variable sets to organize and control the variables used by your devices. This task helps you add, edit, delete, or filter variable sets as needed for device configuration.

Variable sets are used to efficiently manage collections of variables crucial for device configuration. To use variable sets, you must have the IPS license (for Firewall Threat Defense devices) or the Protection license (all other device types).

Procedure

Step 1 Choose **Objects** > **Object Management** > **Variable Set**.

Step 2 Manage your variable sets:

- Add — If you want to add a custom variable set, click **Add Variable Set**; see Create a variable set, on page 121.
- Delete — If you want to delete a custom variable set, click **Delete** ( ) next to the variable set, then click **Yes**. You cannot delete the default variable set or variable sets belonging to ancestor domains.

Note Variables created in a variable set you delete are not deleted or otherwise affected in other sets.

- Edit — If you want to edit a variable set, click **Edit** ( ) next to the variable set you want to modify; see Edit objects, on page 7.
- Filter — If you want to filter variable sets by name, begin entering a name; as you type, the page refreshes to display matching names. If you want to clear name filtering, click **Clear** ( ) in the filter field.
- Manage Variables — To manage the variables included in variable sets, see Manage variables, on page 121.

Create a variable set

Variable sets allow you to define and group variables for use in Secure Firewall Management Center. Use this task when you need to create a new set of variables for policy configuration or reuse.

Procedure

Step 1 Choose **Objects** > **Object Management** > **Variable Set**.

Step 2 Click **Add Variable Set**.

Step 3 Enter a **Name**.

Step 4 (Optional) Enter a **Description**.

Step 5 Manage the variables in the set; see Manage variables, on page 121.

Step 6 Click **Save**.

What to do next

- If an active policy references your object, deploy configuration changes; see Deploy configuration changes.

Manage variables

Configure variable sets to control device policies and manage variables for Secure Firewall Management Center.

- You can display, add, delete, edit, or reset variables as required for your environment.

To manage variables in Secure Firewall Management Center, you must have the IPS license (for Firewall Threat Defense devices) or the Protection license (all other device types).

Procedure

Step 1 Choose **Objects** > **Object Management** > **Variable Set**.

Step 2 Click **Edit** ( ) next to the variable set you want to edit.

If **View** ( ) appears instead, the configuration belongs to an ancestor domain, or you do not have permission to modify the configuration.

Step 3 Manage your variables:

- Display — If you want to display the complete value for a variable, hover your pointer over the value in the **Value** column next to the variable.
- Add — If you want to add a variable, click **Add**; see Add variables, on page 123.
- Delete — Click **Delete** ( ) next to the variable. If you have saved the variable set since adding the variable, click **Yes** to confirm that you want to delete the variable.

  You *cannot* delete these:

  - default variables
  - user-defined variables that are used by intrusion rules or other variables
  - variables belonging to ancestor domains

- Edit — Click **Edit** ( ) next to the variable you want to edit; see Edit variables values in a variable set, on page 124.
- Reset — If you want to reset a modified variable to its default value, click **Reset** next to a modified variable. If reset is dimmed, one of these is true:
  - The current value is already the default value.
  - The configuration belongs to an ancestor domain.

Tip Hover your pointer over an active reset to display the default value.

Step 4 Click **Save** to save the variable set. If the variable set is in use by an access control policy, click **Yes** to confirm that you want to save your changes.

The current value in the default set determines the default value in all other sets. If you modify or reset a variable in the default set, the current value changes in other sets where you have not customized the default value.

What to do next

- If an active policy references your object, deploy configuration changes; see Deploy configuration changes.

Add variables

Add variables to a variable set so you can define reusable network or port values for use in policies.

Use variables in device policies to enable flexible, reusable configurations that simplify network management.

Before you begin

Ensure you have the IPS license.

Procedure

Step 1 In the variable set editor, click **Add**.

Step 2 Enter a unique variable **Name**.

Step 3 From the **Type** drop-down list, choose either **Network** or **Port**.

Step 4 Define the values for the variable:

- To include or exclude specific networks or ports, select them and click **Include** or **Exclude**. If there is overlap, excluded addresses or ports always take precedence over included ones.
- To add literal values, enter a single IP address, address block, port, or port range. If you need to add more than one value, repeat this action.
- To remove any items from included or excluded lists, select the item and click **Delete** ( ) next to the item.
- You may use a combination of literal values, existing variables, objects, or network object groups when specifying included or excluded items for network variables.

Step 5 Save the variable.

- If adding a new variable from a custom set, choose whether to add it as a customized value in the default set or as a default value of `any` in the default and other custom sets.

Step 6 Save the variable set.

- Any access control policy linked to the variable set will display an out-of-date status until changes are deployed.

The variable is added to the variable set and is available for use in policies. Any access control policy linked to the variable set displays an out-of-date status until you deploy the changes.

What to do next

- If an active policy references your object, deploy configuration changes; see Deploy configuration changes.

Edit variables values in a variable set

You must have the IPS license (for Firewall Threat Defense devices) or the Protection license (all other device types).

You can edit both custom and default variables.

You cannot change the **Name** or **Type** values in an existing variable.

Procedure

Step 1 In the variable set editor, click **Edit** ( ) next to the variable you want to modify.

If **View** ( ) appears instead, the object belongs to an ancestor domain, or you do not have permission to modify the object.

Step 2 Modify the variable:

- If you want to move items from the list of available networks or ports to the list of included or excluded items, you can select one or more items and then drag and drop, or click **Include** or **Exclude**.

Tip If addresses or ports overlap between the included and excluded lists for a network or port variable, the excluded addresses or ports take precedence.

- Enter a single literal value, then click **Add**. For network variables, you can enter a single IP address or address block. For port variables, you can add a single port or port range, separating the upper and lower values with a hyphen (-). Repeat this step as needed to enter multiple literal values.
- If you want to remove an item from the included or excluded lists, click **Delete** ( ) next to the item.

Note The list of items to include or exclude can be comprised of any combination of literal strings and existing variables, objects, and network object groups in the case of network variables.

Step 3 Click **Save** to save the variable.

Step 4 Click **Save** to save the variable set. If the variable set is in use by an access control policy, click **Yes** to confirm that you want to save your changes. Your changes are saved, and any access control policy the variable set is linked to displays an out-of-date status.

What to do next

- If an active policy references your object, deploy configuration changes; see Deploy configuration changes.

VLAN tags

A VLAN tag is a network identifier that

- represents a single VLAN or a range of VLAN tags,
- can be grouped with other VLAN tag objects to create tag groups, and

VLAN tag usage

You can use VLAN tag objects and groups in various places in the system's web interface, such as rules and event searches. For example, you could write an access control rule that applies only to a specific VLAN.

Create a VLAN tag object

Use this task when you need to add or manage VLAN tag objects in your Secure Firewall Management Center. VLAN tag objects are referenced in access control and other policies to match specific VLAN tags or ranges.

Procedure

Step 1 Choose **Objects** > **Object Management** > **VLAN Tag**.

Step 2 Choose **Add Object** from the **Add VLAN Tag** drop-down list.

Step 3 Enter a **Name**.

Step 4 Enter a **Description**.

Step 5 Enter a value in the **VLAN Tag** field. Use a hyphen to specify a range of VLAN tags.

Step 6 Manage overrides for the object:

- If you want to allow overrides for this object, check the **Allow Overrides** check box; see Allow object overrides, on page 14.
- If you want to add override values to this object, expand the Override section and click **Add**; see Add object overrides, on page 14.

Step 7 Click **Save**.

What to do next

- If an active policy references your object, deploy configuration changes; see Deploy configuration changes.

VPN

You can use the following VPN objects on Firewall Threat Defense devices. To use these objects, you must have Admin privileges, and your Smart License account must satisfy export controls. You can configure these objects in leaf domains only.

Certificate map objects

A certificate map object is a configuration element that

- contains a named set of certificate matching rules,
- associates a received certificate with a remote access VPN connection profile, and
- matches rules in priority order, ending when the first rule results in a match.

Certificate map object reference information

Remote access VPN policies use certificate map objects to map certificates to connection profiles. If a received certificate matches the rules in a certificate map, the system associates the connection with the specified profile.

Both certificate map objects and connection profiles are components of a remote access VPN policy.

- The rules are matched in the order shown in the UI.
- Matching ends when the first rule within the certificate map object results in a match.

Navigation path for certificate map objects:

- **Objects** > **Object Management** > **VPN** > **Certificate Map**

Fields in certificate map objects:

- **Name**: Identifies this object so it can be referred to from other configurations, such as remote access VPN.
- **Mapping criteria**: Specifies the contents of the certificate to evaluate. If the certificate satisfies these rules, the user is mapped to the connection profile containing this object.
  - **Field**: Select the field for the matching rule according to the subject or issuer of the client certificate.

    If you set the **Field** to *Alternative Subject* or *Extended Key Usage*, the component remains fixed as *Whole Field*.
  - **Component**: Select the component of the client certificate to use for the matching rule.



Note **SER (Serial Number) component**: Ensure you specify the serial number for the subject field. The certificate map only matches with a serial number attribute in the subject name.

  - **Operator**: Select the operator for the matching rule:
    - Equals: The certificate component must match the entered value. If they do not match exactly, the connection is denied.
    - Contains: The certificate component must contain the entered value. If the component does not contain the value, the connection is denied.

- Does not equal: The certificate component cannot equal the entered value. For example, for a selected certificate component of country and an entered value of US, if the client country value equals US, then the connection is denied.
- Does not contain: The certificate component cannot contain the entered value. For example, for a selected certificate component of country and an entered value of US, if the client country value contains US, the connection is denied.

- **Value**: The value of the matching rule. The value entered is associated with the selected component and operator.

Related Topics

Configure Certificate Maps

Secure Client custom attributes objects

Custom attributes objects enable Secure Client (formerly AnyConnect) to configure features such as Per App VPN, allow or defer upgrade, and dynamic split tunneling. You define the attribute type, then assign one or more named values for this type. In Secure Firewall Management Center, you create custom attributes objects, add them to a group policy, and associate the policy with a remote access VPN to enable features for VPN clients.

Supported features using custom attributes objects

Firewall Threat Defense supports these features using custom attribute objects:

- **Per App VPN**—You can specify which applications may use the VPN tunnel. The system identifies these apps and tunnels only the allowed applications.
- **Allow or defer upgrade**—You can delay downloading Secure Client updates. When a client update becomes available, you can prompt users to upgrade or defer the update.
- **Dynamic Split Tunneling**—You can configure policies to selectively include or exclude IP addresses or networks from the VPN tunnel. Create a custom attribute and add it to a group policy to achieve this.

For step-by-step instructions to configure Secure Client custom attributes, see Configure custom attributes for Secure Client, on page 127.

For details about the specific custom attributes to configure for a feature, see the *Cisco Secure Client (including AnyConnect) Administrator Guide* for the Secure Client release you are using.

Configure custom attributes for Secure Client

Use this task when you need to configure custom attributes for Secure Client clients, such as Per App VPN, update deferment, or dynamic split tunneling, in Secure Firewall Management Center.

This is relevant for environments where device management and application-specific VPN rules are required, and when deploying secure client features that depend on custom attribute objects.

Applicable when integrating MDM-enrolled devices and advanced VPN features.

Before you begin

Ensure that you have done these steps before adding a custom attribute object for Per App VPN:

- Per App VPN must be properly configured via MDM and each device must be enrolled to the MDM server.
- Create a base64 encoded string for each app using the Cisco Secure Client Enterprise Application Selector Tool.
  1. Download the Cisco Secure Client Enterprise Application Selector Tool from here.
  2. Open the Application Selection Tool and select the mobile platform from the drop-down menu located on the upper left.
  3. Add rule by entering friendly name and App ID; the other fields are optional.
  4. On the menu bar, click **Policy**. The encoded base64 rule appears.
  5. Select and copy the policy string, and save it to use later when you create the Secure Client Custom Attributes object.

Procedure

Step 1 Choose **Objects** > **Object Management** > **VPN** > **Custom Attribute**.

Step 2 Click **Secure Client Custom Attribute**.

Step 3 Enter a **Name** and optionally a **Description** for the attribute.

Step 4 Select an attribute from the **Secure Client Attribute** drop-down list:

- **Per App VPN**—Select this option and specify the base64 encoded string in the **Attribute Value** box.
- **Allow Defer Update**—Select one of these options and specify the required information to allow or defer Secure Client update:
  - **Show the prompt until user takes action**—Display the prompt to the VPN user until the user chooses to allow or defer the VPN client update.
  - **Show the prompt until times out**—Choose this option to display the prompt for a specified duration and specify the duration in the **Timeout** box.
  - **Do not show the prompt and take automatic action**—Choose this option to automatically allow or defer the VPN update.
  - **Default Action**—Select the default action to be taken when the user does not respond, or when you want to configure an automatic action without the user's intervention. You can choose to update the Secure Client or postpone the update.
  - **Minimum Version**—Specify the minimum Secure Client version to be present on the client system to allow or defer the update.
- **Dynamic Split Tunneling**—Select this option to include or exclude IP addresses or networks from the VPN tunnel.
  - **Include domains**—Specify domain names that will be included in the remote access VPN tunnel.
  - **Exclude domains**—Specify domain names that will be excluded from the remote access VPN tunnel.

Step 5 Select the **Allow Overrides** check box to enable object overrides.

Step 6 Click **Save**.
The custom attributes object is added to the list.

What to do next

Associate the custom attributes with a group policy. See Add custom attributes to your group policy, on page 129.

Add custom attributes to your group policy

Add custom attributes to a group policy to enable their use with remote access VPN connections. You can assign advanced VPN features to users based on their group policy membership.

To use Secure Client custom attributes with remote access VPN connections, you must associate them with the appropriate group policy. This enables you to offer advanced VPN customization to users based on their group policy membership.

Procedure

Step 1 Select **Objects** > **Object Management** > **VPN** > **Group Policy**.

Step 2 Add a new group policy or edit an existing group policy.

Step 3 Click **Secure Client** > **Custom Attributes**.

Step 4 Click **Add**.

Step 5 Select the **Secure Client Attribute**: Per App VPN, Allow Defer Update, or Dynamic Split Tunneling.

Step 6 Select a **Custom Attribute Object** from the list.

Note Click Add (+) to create a new custom attribute object for the selected Secure Client attribute. You can also create a custom attribute object at **Objects** > **Object Management** > **VPN** > **Custom Attribute**. See Configure custom attributes for Secure Client, on page 127.

Step 7 Click **Add** to save the attributes to the group policy, and then click **Save** to save the changes to the group policy.

Related Topics

Group policy options for Secure Client, on page 133

Firewall Threat Defense group policy objects

A group policy is a set of attribute and value pairs, stored in a group policy object, that define the remote access VPN experience. For example, in the group policy object, you configure general attributes such as addresses, protocols, and connection settings.

Group policy object assignment and licensing

The group policy applied to a user is determined when the VPN tunnel is established. The RADIUS authorization server assigns the group policy, or it is obtained from the current connection profile.

Note There is no group policy attribute inheritance on the Firewall Threat Defense. A group policy object is used, in its entirety, for a user. The group policy object identified by the AAA server upon login is used, or, if that is not specified, the default group policy configured for the VPN connection is used. The provided default group policy can be set to your default values, but will only be used if it is assigned to a connection profile and no other group policy has been identified for the user.

To use group objects, you must have one of these Secure Client licenses associated with your Smart License account with Export-Controlled Features enabled:

- Secure Client VPN Only
- Secure Client Advantage
- Secure Client Premier

Configure a group policy object

Group policy objects are used to manage and enforce security and access policies for remote access VPN connections.

See Firewall Threat Defense group policy objects, on page 129.

Procedure

Step 1 Choose **Objects** > **Object Management** > **VPN** > **Group Policy**.

Previously configured policies are listed including the system default. Depending on your level of access, you may edit, view, or delete a group policy.

Step 2 Click **Add Group Policy** or choose a current policy to edit.

Step 3 Enter a **Name** and optionally a **Description** for this policy.

The name can be up to 64 characters. Spaces are allowed. The description can be up to 1,024 characters.

Step 4 Specify the **General** parameters for this Group Policy as described in Group policy general options, on page 131.

Step 5 Specify the **Secure Client** parameters for this Group Policy as described in Group policy options for Secure Client, on page 133.

Step 6 Specify the **Advanced** parameters for this Group Policy as described in Advanced group policy option fields and values, on page 136.

Step 7 Click **Save**.
The new Group Policy is added to the list.

What to do next

Add the group policy object to a remote access VPN connection profile.

Group policy general options

This topic describes the general options available when configuring a group policy, including navigation, VPN protocol selection, IP address pool assignment, banner configuration, DNS/WINS settings and split tunneling options.

Navigation path

Objects > **Object Management** > **VPN** > **Group Policy**, click **Add Group Policy** or choose an existing policy to edit, then select the **General** tab.

VPN protocols fields

Specify the types of Remote Access VPN tunnels (**SSL** or **IPsec IKEv2**) you want to apply for this group policy.

IP address pools

- Specifies the IPv4 address assignment based on address pools designated for user groups in Remote Access VPN.
- You can assign IP addresses from specific pools for identified user groups using RADIUS/ISE for authorization.
- You can enforce policy for users or user groups in non-identity-aware systems by configuring the Group Policy as the RADIUS Authorization attribute (GroupPolicy/Class).

Example: Select a specific address pool for contractors and enforce policy for restricted access to the internal network using those addresses.

The Firewall Threat Defense device assigns IPv4 address pools to the clients in this order:

1. RADIUS attribute for IPv4Address Pool
2. RADIUS attribute for Group Policy
3. Address Pool in Group Policy mapped to a Connection Profile
4. IPv4Address Pool in Connection Profile

Limitations:

- IPv6 address pool is not supported.
- A maximum of six IPv4 address pools can be configured in a Group Policy.
- Deployment failures may occur when address pools in use are modified. Before making any changes to the address pools, you must log off all users.
- If you rename address pools or configure overlapping address pools, deployment may fail. To address this, first remove the old address pool, then deploy the changed address pool.

  Troubleshooting commands :

- `show ip local pool <address-pool-name>`
- `show vpn-sessiondb detail anyconnect`
- `vpn-sessiondb loggoff all noconfirm`

Banner fields

Specifies the banner text to present to users at login. You can enter up to 491 characters in the banner. By default, no value is configured. The IPsec VPN client supports full HTML for the banner, however, the Secure Client supports only partial HTML. To ensure that the banner displays properly to remote users, use the /n tag for IPsec clients and the <BR> tag for SSL clients.

DNS/WINS fields

Domain Naming System (DNS) and Windows Internet Naming System (WINS) servers. Used for Secure Client name resolution.

- **Primary DNS Server** and **Secondary DNS Server**—Choose or create a Network Object which defines the IPv4 or IPv6 addresses of the DNS servers you want this group to use.
- **Primary WINS Server** and **Secondary WINS Server**—Choose or create a Network Object containing the IP addresses of the WINS servers you want this group to use.
- **DHCP Network Scope**—Choose or create a Network Object containing a routable IPv4 address on the same SUBNET as the desired pool, but not within the pool. The DHCP server determines which SUBNET this IP address belongs to and assigns an IP address from that pool. If not set properly, deployment of the VPN policy fails.

  If you configure DHCP servers for the address pool in the connection profile, the DHCP scope identifies the subnets to use for the pool for this group. The DHCP server must also have addresses in the same SUBNET identified by the scope. The scope allows you to select a subset of the address pools defined in the DHCP server for this specific group.

  If you do not define a network scope, the DHCP server assigns IP addresses in the order of the address pools configured. It goes through the pools until it identifies an unassigned address.

  We recommend using the IP address of an interface whenever possible for routing purposes. For example, if the pool is 10.100.10.2-10.100.10.254, and the interface address is 10.100.10.1/24, use 10.100.10.1 as the DHCP scope. Do not use the network number. Use DHCP only for IPv4 addressing. If the address you choose is not an interface address, you might need to create a static route for the scope address.

  LINK-SELECTION (RFC 3527) and SUBNET-SELECTION (RFC 3011) are currently not supported.
- **Default Domain**—Name of the default domain. Specify a top-level domain, for example, example.com.

Split tunneling fields

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

- **IPv4 Split Tunneling / IPv6 Split Tunneling**—By default, split tunneling is not enabled. For both IPv4 and IPv6 it is set to **Allow all traffic over tunnel**. Left as is, all traffic from the endpoint goes over the VPN connection.

To configure split tunneling, choose the **Tunnel networks specified below** or **Exclude networks specified below** policy, and then configure an access control list for that policy.

- **Split Tunnel Network List Type**—Choose the type of Access List you are using. Then choose or create a **Standard Access List** or **Extended Access List**. See Access lists, on page 21 for details.

Note From Version 7.4, the **Standard Access List** option is disabled by default. Use the **Extended Access List** option to add an access list.

- **DNS Request Split Tunneling**—Also known as Split DNS. Configure the DNS behavior expected in your environment.

  By default, split DNS is not enabled and set to **Send DNS request as per split tunnel policy**. Choosing **Always send DNS request over tunnel** forces all DNS requests to be sent over the tunnel to the private network.

  To configure split DNS, choose **Send only specified domains over tunnel**, and enter the list of domain names in the **Domain List** field. These requests are resolved through the split tunnel to the private network. All other names are resolved using the public DNS server. Enter up to 10 entries in the list of domains, separated by commas. The entire string can be no longer than 491 characters.

Related Topics

Configure a group policy object, on page 130

Group policy options for Secure Client

This topic lists and describes the specifications and configuration fields for group policy Secure Client VPN operation. This includes navigation steps, profile fields, management profile settings, client modules, SSL settings, connection options, and custom attributes.

These specifications apply to the operation of the Secure Client VPN.

Navigation

Objects > **Object Management** > **VPN** > **Group Policy**. Click **Add Group Policy** or choose a current policy to edit., and then select the **Secure Client** tab.

Profile fields

Profile—Choose or create a file object containing the Secure Client Profile. See File objects, on page 143 for object creation details.

The Secure Client Profile is a group of configuration parameters stored in an XML file. The Secure Client software uses it to configure the connection entries that appear in the client's user interface. These parameters (XML tags) also configure settings to enable more Secure Client features.

Use the GUI-based Secure Client Profile Editor, an independent configuration tool, to create the Secure Client Profile. See the *Secure Client Profile Editor* chapter in the appropriate release of the Cisco Secure Client (including AnyConnect) Administrator Guide for details.

Management profile fields

Management VPN Tunnel—Provides always-on connectivity to the corporate network when the endpoint is powered up, regardless of the user's VPN connection status.

Management VPN Profile—The Management Profile file contains settings for enabling and establishing Management VPN Tunnel on endpoint.

The Standalone Management VPN Tunnel profile editor can be used to create a new profile file or modify an existing file. You can download the profile editor from Cisco Software Download Center.

For more information about adding a profile file, see File objects, on page 143.

Client modules fields

Cisco Secure Client VPN Only offers enhanced security through various built-in modules. These modules provide services such as web security, network visibility into endpoint flows, and off-network roaming protection. Each module includes a custom client profile.

The following Secure Client modules are optional and you can configure these modules to be downloaded when a VPN user downloads Secure Client:

- **AMP Enabler**—Deploys advanced malware protection (AMP) for endpoints.
- **DART**—Captures a snapshot of system logs and other diagnostic information, which can be sent to the Cisco TAC for troubleshooting.
- **ISE Posture**—Uses the OPSWAT library to perform posture checks to assess an endpoint's compliance.
- **Network Access Manager**—Provides 802.1X (Layer 2) and device authentication for access to both wired and wireless networks.
- **Network Visibility**—Enhances the enterprise administrator's ability to do capacity and service planning, auditing, compliance, and security analytics.
- **Start Before Login**—Forces the user to connect to the enterprise infrastructure over a VPN connection before logging on to Windows by starting Secure Client before the Windows login dialog box appears.
- **Umbrella Roaming Security**—Provides DNS-layer security when no VPN is active.
- **Web Security**—Enforces web security policies and blocks malicious content.

Click **Add** and select the following for each client module:

- **Client Module**—Select the Secure Client module from the list.
- **Profile to download**—Choose or create a file object containing the Secure Client Profile. See File objects, on page 143 for object creation details.
- **Enable module download**—Select to enable endpoints to download the client module along with the profile. If not selected, the endpoints can download only the client profile.

Use the GUI-based Secure Client Profile Editor, an independent configuration tool to create a client profile for each module. Download the Secure Client Profile Editor from Cisco Software Download Center. See the *Secure Client Profile Editor* chapter in the appropriate release of the Cisco Secure Client (including AnyConnect) for details.

SSL settings fields

- **SSL Compression**—Determines if data compression is enabled, and specifies the method to use (Deflate or LZS). SSL Compression is disabled by default.

  Data compression speeds up transmission rates, but also increases the memory requirement and CPU usage for each user session. Thereby, decreasing the overall throughput of the security appliance.

- **DTLS Compression**—Specifies whether to compress Datagram Transport Layer Security (DTLS) connections for this group using LZS. DTLS Compression is disabled by default.

- **MTU Size**—TSpecifies the maximum transmission unit (MTU) size for SSL VPN connections established by Cisco Secure Client VPN Only. The default is 1406 bytes; the valid range is 576 to 1462 bytes.

  - **Ignore DF Bit**—Whether to ignore the Don't Fragment (DF) bit in packets that need fragmentation. Allows the forced fragmentation of packets that have the DF bit set, allowing them to pass through the tunnel.

Connection settings fields

- **Enable Keepalive Messages between Secure Client and VPN gateway** and its **Interval** setting.—Determines whether peers exchange keepalive messages to demonstrate availability for sending and receiving data in the tunnel. By default, this feature is enabled. Keepalive messages are transmitted at set intervals. If enabled, enter the time interval (in seconds) that the remote client waits between sending IKE keepalive packets. The default interval is 20 seconds, the valid range is 15 to 600 seconds.

- **Enable Dead Peer Detection** and its **Interval** settings.—Dead Peer Detection (DPD) ensures that the VPN secure gateway or client quickly detects when the peer is no longer responding and the connection has failed. Default is enabled for both the gateway and the client. DPD messages transmit at set intervals. If enabled, enter the time interval (in seconds) that the remote client waits between sending DPD messages. The default interval is 30 seconds, the valid range is 5 to 3600 seconds.

- **Enable Client Bypass Protocol**—Allows you to configure how the secure gateway manages IPv4 traffic (when it is expecting only IPv6 traffic), or how it manages IPv6 traffic (when it is expecting only IPv4 traffic).

  After the Secure Clientconnects to the headend, the headend assigns the VPN connection an IPv4 address, an IPv6 address, or both. If the headend assigns only one type of address, you can configure the Client Bypass Protocol to either drop network traffic for which an IP address was not assigned (default, disabled, not checked), or allow that traffic to bypass the headend and be sent from the client unencrypted (“in the clear”; enabled, checked).

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

- **SSL rekey**—Enables the client to rekey the connection, renegotiating the crypto keys and initialization vectors, increasing the security of the connection. This is disabled by default. When enabled, the renegotiation can be done at a specified interval and rekey the existing tunnel or create a new tunnel by setting the following fields:

  - **Method**—Available when SSL rekey is enabled. Create a **New Tunnel** (default), or renegotiate, the **Existing Tunnel**'s specifications.

- **Interval**—Available when SSL rekey is enabled. Set to a default of 4 minutes with a range of 4-10080 minutes (1 week).

- **Client Firewall Rules**—Use the Client Firewall Rules to configure firewall settings for the VPN client's platform. Rules are based on criteria such as source address, destination address, and protocol. Extended Access Control List building block objects are used to define the traffic filter criteria. Choose or create an Extended ACL for this group policy. Define a **Private Network Rule** to control data flowing to the private network, a **Public Network Rule** to control data flowing "in the clear", outside of the established VPN tunnel, or both.



Note Ensure that the ACL contains only TCP/UDP/ICMP/IP ports and source network as any, any-ipv4 or any-ipv6.

Only VPN clients running Microsoft Windows can use these firewall settings.

Custom attributes fields

Custom attributes are used by the Secure Client to configure features such as Per App VPN, Allow or defer upgrade, and Dynamic split tunneling. Click **Add** to add custom attributes to the group policy.

1. Select the **Secure Client Attrinute**: Per App VPN, Allow Defer Update, or Dynamic Split Tunneling.
2. Select a **Custom Attribute Object** from the list.



Note Click Add (+) to create a new custom attribute object for the selected Secure Client attribute. You can also create a custom attribute object at **Objects** > **Object Management** > **VPN** > **Custom Attribute**. See Configure custom attributes for Secure Client, on page 127.

3. Click **Add** to save the attributes to the group policy, and then click **Save** to save the changes to the group policy.

Related Topics

Configure a group policy object, on page 130

Advanced group policy option fields and values

The advanced group policy includes options and attributes for configuring VPN access and session behavior. This reference describes where to locate advanced settings, outlines attributes related to traffic filtering, and summarizes session control fields and their allowed values.

Navigation path

Objects > **Object Management** > **VPN** > **Group Policy**. Click **Add Group Policy** or choose a current policy to edit. Then select the **Advanced** tab.

Traffic filter fields

- **Access List Filter**—Filters consist of rules that determine whether to allow or block tunneled data packets coming through the VPN connection. You set rules using criteria such as source address, destination address, and protocol.

  The VPN filter applies only to initial connections. It does not control secondary connections, such as a SIP media connection, that are opened during application inspection.

  Use Extended Access Control List building block objects to define traffic filter criteria. Choose or create an Extended ACL for this group policy.

- **Restrict VPN to VLAN**—Also called "VLAN mapping," this parameter specifies the egress VLAN interface for sessions to which this group policy applies. The ASA forwards all group traffic to the VLAN you select.

  Use this attribute to assign a VLAN to the group policy to simplify access control. You can assign a value to this attribute instead of using ACLs to filter session traffic. The drop-down list shows only the VLANs that are configured in this ASA. Allowed values range from 1 to 4094. The default value is Unrestricted.

Session settings fields

- **Access Hours**—Choose or create a time range object. This object specifies the range of time this group policy is available to be applied to a remote access user. See Time ranges, on page 104 for details.
- **Simultaneous Logins Per User**—Specifies the maximum number of simultaneous logins allowed for a user. The default value is 3. If the minimum value is set to 0, logins are disabled and users cannot access the VPN. If you allow several simultaneous connections, security may be compromised and performance reduced.
- **Maximum Connection Time / Alert Interval**—Specifies the maximum user connection time in minutes. At the end of this time, the system disconnects the user. The minimum is 1 minute). The Alert interval lets you display a message to the user before the maximum connection time is reached.
- **Idle Timeout / Alert Interval**—Specifies this user's idle timeout period in minutes. If there is no communication activity on the user connection in this period, the system stops the connection. The minimum time is 1 minute. The default is 30 minutes. The Alert interval lets you display a message to the user before the idle time is reached.

Related Topics

Configure a group policy object, on page 130

Firewall Threat Defense IPsec proposals

IPsec Proposals (or Transform Sets) are used when configuring VPN topologies. During the IPsec security association negotiation with ISAKMP, both peers must agree on the same proposal to protect the data flow.

IPsec proposal types and negotiation

IPsec proposals are categorized based on the IKE version: IKEv1 or IKEv2.

- For an IKEv1 IPsec proposal (Transform Set), select the mode for IPsec operation and define the required encryption and authentication types. Select only one option for each algorithm. To support multiple combinations in a VPN, create multiple IKEv1 IPsec proposal objects.

- For an IKEv2 IPsec proposal, select all encryption and hash algorithms allowed in the VPN. During IKEv2 negotiations, the peers negotiate and use the most appropriate options supported by both devices.

Both IKEv1 and IKEv2 IPsec proposals use the Encapsulating Security Protocol (ESP), which provides authentication, encryption, and antireplay services. ESP uses IP protocol type 50.

Note We recommend that you use both encryption and authentication on IPsec tunnels.

Configure an IKEv1 IPsec proposal object

Use this task when you need to create or modify IKEv1 IPsec proposal objects in Secure Firewall Management Center. Proposal objects define the cryptographic parameters for VPN connections and are required for establishing secure tunnels between firewalls or security gateways.

Procedure

Step 1 Choose **Objects** > **Object Management** > **VPN** > **IKEv1 IPsec Proposal** and then **VPN** > **IPsec IKev1 Proposal** from the table of contents.

Previously configured Proposals are listed including system defined defaults. Depending on your level of access, you may **Edit** ( ), **View** ( ), or **Delete** ( ) a Proposal.

Step 2 Choose **Add IPsec IKEv1 Proposal** to create a new Proposal.

Step 3 Enter a **Name** for this proposal.

The name of the policy object can be up to 128 characters.

Step 4 Enter a **Description** for this Proposal.

Provide a description of the policy object that is up to 1024 characters.

Step 5 Choose the **ESP Encryption** method. The Encapsulating Security Protocol (ESP) encryption algorithm for this Proposal.

For IKEv1, select one of the options. When deciding which encryption and Hash Algorithms to use for the IPsec proposal, your choice is limited to algorithms supported by the devices in the VPN. For a full explanation of the options, see Decide encryption algorithms for VPN policies.

Step 6 Select an option for **ESP Hash**.

For a full explanation of the options, see Decide which hash algorithms to use.

Step 7 Click **Save**
The new Proposal is added to the list.

Configure an IKEv2 IPsec proposal object

Use this task when you need to define or update the cryptographic settings for IKEv2 IPsec VPN tunnels on your Secure Firewall Management Center. This is relevant when setting up new VPNs or adjusting security policies to meet organizational requirements.

Procedure

Step 1 Choose **Objects** > **Object Management** > **VPN** > **IKEv2 IPsec Proposal** from the table of contents.

Previously configured proposals are listed including system defined defaults. Depending on your level of access, you may **Edit** ( ), **View** ( ), or **Delete** ( ) a Proposal.

Step 2 Choose **Add IKEv2 IPsec Proposal** to create a new proposal.

Step 3 Enter a **Name** for this proposal.

The policy object name can be up to 128 characters.

Step 4 Enter a **Description** for this proposal.

The policy object description can contain up to 1,024 characters.

Step 5 Click **ESP Hash**, the hash or integrity algorithm to use in the proposal for authentication.

Note Firewall Threat Defense does not support IPSec tunnels with NULL encryption. Make sure that you do not choose NULL encryption for IPSec IKEv2 proposal.

For IKEv2, select all the options you want to support for **ESP Hash**. For a full explanation of the options, see Decide which hash algorithms to use.

Step 6 Click **ESP Encryption** . Select the Encapsulating Security Protocol (ESP) encryption algorithm for this proposal.

For IKEv2, click Select to open a dialog box where you can select all of the options you want to support. When deciding which encryption and hash algorithms to use for the IPsec proposal, choose only algorithms supported by the devices in the VPN. For a full explanation of the options, see Decide encryption algorithms for VPN policies.

Step 7 Click **Save**.
The new proposal is added to the list.

IKE policies

A IKE policy is a key management policy that

- defines security parameters and algorithms used to authenticate IPsec peers,
- negotiates and distributes IPsec encryption keys, and

- establishes secure communications by negotiating security associations in IPsec VPNs.
- The Internet Key Exchange (IKE) protocol manages authentication, key negotiation, and automatic establishment of IPsec security associations (SAs).
- IKE negotiation occurs in two phases:
  - Phase 1 negotiates a security association between two IKE peers, enabling secure communication in Phase 2.
  - Phase 2 establishes SAs for applications, such as IPsec.
- Both phases use proposals to negotiate connections. An IKE proposal is a set of algorithms that two peers use to secure negotiation.
- IKE negotiation starts with peers agreeing on a shared IKE policy, which defines the security parameters to protect future negotiations.
- For IKEv1, proposals include a single set of algorithms and a modulus group; multiple, prioritized policies can be created to match remote peers.
- For IKEv2, you can select multiple algorithms and modulus groups in one policy.
- Peers choose proposals during Phase 1 negotiation, enabling creation of a single IKE proposal but consideration of multiple prioritized options.
- For IKEv2, the policy object does not specify authentication; other policies must define authentication requirements.
- An IKE policy is required when you configure a site-to-site IPsec VPN. .

Configure an IKEv1 policy object

Use the IKEv1 Policy page to create, edit, or delete IKEv1 policy objects. These policy objects specify settings such as encryption, authentication, priority, and other parameters necessary for establishing secure VPN tunnels.

Procedure

Step 1 Choose **Objects** > **Object Management** > **VPN** > **IKEv1 Policy** from the table of contents.

Previously configured policies are listed including system-defined defaults. Depending on your level of access, you may **Edit** (), **View** (), or **Delete** ( ) a proposal.

Step 2 (Optional) Click **Add IKEv1 Policy** to create a new policy object. Then, enter a **Name** for this policy. The policy object name can be up to 128 characters. Optionally, you can enter the description. The policy object description can contain up to 1,024 characters.

Step 3 Enter the **Priority** value of the IKE policy.

The priority value determines the order of the IKE policy compared by the two negotiating peers when attempting to find a common security association (SA). If the remote IPsec peer does not support the parameters selected in your first priority policy, it tries to use the parameters defined in the next lowest priority. Valid values range from 1 to 65,535. The lower the number, the higher the priority. If you leave this field blank,

Management Center assigns the lowest unassigned value starting with 1, then 5, then continuing in increments of 5.

Step 4 Select the **Encryption** method.

When deciding which encryption and Hash Algorithms to use for the IKEv1 policy, your choice is limited to algorithms supported by the peer devices. For an extranet device in the VPN topology, you must choose the algorithm that matches both peers. For IKEv1, select one of the options. For a full explanation of the options, see Decide encryption algorithms for VPN policies.

Step 5 Select the **Hash** Algorithm that creates a Message Digest, which is used to ensure message integrity.

When deciding which encryption and Hash Algorithms to use for the IKEv1 proposal, your choice is limited to algorithms supported by the managed devices. For an extranet device in the VPN topology, you must choose the algorithm that matches both peers. For a full explanation of the options, see Decide which hash algorithms to use.

Step 6 Select the **Diffie-Hellman Group**.

The Diffie-Hellman group to use for encryption. A larger modulus provides higher security but requires more processing time. The two peers must have a matching modulus group. Select the group that you want to allow in the VPN. For a full explanation of the options, see Decide which Diffie-Hellman modulus group to use.

Step 7 Enter the **Lifetime** of the security association (SA), in seconds. You can specify a value from 120 to 2,147,483,647 seconds. The default is 86400.

When the lifetime is exceeded, the SA expires and must be renegotiated between the two peers. Generally, the shorter the lifetime (up to a point), the more secure your IKE negotiations. However, with longer lifetimes, future IPsec security associations can be set up more quickly than with shorter lifetimes.

Step 8 Select the **Authentication Method** to use between the two peers.

- **Preshared Key**—Preshared keys allow for a secret key to be shared between two peers and to be used by IKE during the authentication phase. If one of the participating peers is not configured with the same preshared key, the IKE SA cannot be established.
- **Certificate**—When you use Certificates as the authentication method for VPN connections, peers obtain digital certificates from a CA server in your PKI infrastructure, and trade them to authenticate each other.

Note In a VPN topology that supports IKEv1, the **Authentication Method** specified in the chosen IKEv1 Policy object becomes the default in the IKEv1 **Authentication Type** setting. These values must match, otherwise, your configuration will error.

Step 9 Click **Save**
The new IKEv1 policy is added to the list.

Configure an IKEv2 policy object

Use the IKEv2 policy dialog box to create, delete, and edit an IKEv2 policy object. These policy objects contain the parameters required for IKEv2 policies.

Procedure

Step 1 Choose **Objects** > **Object Management** > **VPN** > **IKEv2 Policy** from the table of contents.

Previously configured policies are listed including system defined defaults. Depending on your level of access, you may **Edit** (), **View** (), or **Delete** ( ) a policy.

Step 2 Click **Add IKEv2 Policy** to create a new policy. Then, enter a **Name** for this policy. The policy object name can be up to 128 characters. Optionally, enter the description. The description can be up to 1024 characters.

Step 3 Enter the **Priority**.

The priority value of the IKE proposal. The priority value determines the order of the IKE proposals compared by the two negotiating peers when attempting to find a common security association (SA). If the remote IPsec peer does not support the parameters selected in your first priority policy, it tries to use the parameters defined in the next lowest priority policy. Valid values range from 1 to 65535. The lower the number, the higher the priority. If you leave this field blank, Management Center assigns the lowest unassigned value starting with 1, then 5, then continuing in increments of 5.

Step 4 Enter the **Lifetime** of the security association (SA), in seconds. You can specify a value from 120 to 2,147,483,647 seconds. The default is 86400.

When the lifetime is exceeded, the SA expires and must be renegotiated between the two peers. Generally, the shorter the lifetime (up to a point), the more secure your IKE negotiations. However, with longer lifetimes, future IPsec security associations can be set up more quickly than with shorter lifetimes.

Step 5 Select algorithms for these methods:

a. Click **Integrity Algorithms** portion of the Hash Algorithm used in the IKE policy. The Hash Algorithm creates a Message Digest, which is used to ensure message integrity.

When deciding which encryption and Hash Algorithms to use for the IKEv2 proposal, your choice is limited to algorithms supported by the managed devices. For an extranet device in the VPN topology, you must choose the algorithm that matches both peers. Select all the algorithms that you want to allow in the VPN.For a full explanation of the options, see Decide which hash algorithms to use.

b. Click **Encryption Algorithm** used to establish the Phase 1 SA for protecting Phase 2 negotiations.

When deciding which encryption and Hash Algorithms to use for the IKEv2 proposal, your choice is limited to algorithms supported by the managed devices. For an extranet device in the VPN topology, you must choose the algorithm that matches both peers. Select all the algorithms that you want to allow in the VPN. For a full explanation of the options, see Decide encryption algorithms for VPN policies.

c. Click **PRF Algorithm**.

The pseudorandom function (PRF) portion of the Hash Algorithm used in the IKE policy. In IKEv1, the Integrity and PRF algorithms are not separated, but in IKEv2, you can specify different algorithms for these elements. Select all of the algorithms that you want to allow in the VPN. For a full explanation of the options, see Decide which hash algorithms to use.

d. Click **Diffie-Hellman Group** for encryption.

A larger modulus provides higher security but requires more processing time. The two peers must have a matching modulus group. Select the groups that you want to allow in the VPN. For a full explanation of the options, see Decide which Diffie-Hellman modulus group to use.

Step 6 Click **Save**.
If a valid combination of choices has been selected the new IKEv2 policy is added to the list. If not, errors are displayed and you must make changes accordingly to successfully save this policy.

Secure client customizations

A Secure Client customization is a configuration object that

- represents files used to customize Secure Client,
- enables deployment of customizations to the VPN headend, and
- allows threat defense to distribute customizations to endpoints when users connect from Secure Client.

Supported secure client customizations

Secure Client customization objects represent files used to modify Secure Client. Secure Client supports these types of customizations:

- GUI text and messages
- Icons and images
- Scripts
- Binaries
- Custom installer transforms
- Localized installer transforms

For more information about configuring Secure Client customizations, see Customize Cisco Secure Client.

File objects

File objects are configuration element that

- represent files used in configurations, typically for remote access VPN policies
- can contain Secure Client Profile and Secure Client Image files, and
- are created, edited, and managed through the Add and Edit File Object dialog boxes.

File object management and usage

File objects are used to manage files for remote access VPN policies and are associated with profiles and images for deployment to endpoints.

File objects can be created and edited using the Add and Edit File Object dialog boxes. They are used in configurations for remote access VPN policies and can contain Secure Client Profile and Secure Client Image files.

Create profiles for each AnyConnect module and Secure Client Management VPN using independent profile editors. Deploy these profiles to meet administrator-defined end user requirements and authentication policies on endpoints Secure Client only. This makes preconfigured network profiles available to end users

When you create a file object, the Firewall Management Center makes a copy of the file in its repository. When you back up the database, these files are included. If you restore the database, the files are restored as well. To use a file in a file object, upload it through the management center interface, not directly to the file repository.

When you deploy configurations that specify a file object, the associated file is downloaded to the device in the appropriate directory.

You can take these actions for each file object:

- **Download**: Download the Secure Client file.
- **Edit**: Modify the file object details.
- **Delete**: Delete the Secure Client file object. When you delete a file object, the associated file is not deleted from the file repository, only the object is deleted.

Navigation path

Objects > **Object Management** > **VPN** > **Secure Client File**.

Fields

- **Name**: Enter the name of the file to identify the file object; you can add up to 128 characters.
- **File Name**: Click **Browse** to select the file. The file name and full path of the file are added when you select the file.
- **File Type**: Choose the file type corresponding to the file you have selected. These file types are available:

File Types	Details
Secure Client Image	Select when you add the Secure Client image you have downloaded from the Cisco Software Download Center. You can associate any new or additional Secure Client images to the remote access VPN policy. You can also remove unsupported or end-of-life client packages that are no longer required.
Secure Client VPN Profile	Choose for the Secure Client VPN profile file. The profile file is created using the GUI-based Secure Client Profile Editor, an independent configuration tool. See the *Secure Client Profile Editor* chapter in the appropriate release of the Cisco Secure Client (including AnyConnect) User Guide for details.
Secure Client Management VPN Profile	Select when you add a profile file for the Secure Client management VPN tunnel. Download the Secure Client **VPN Management Tunnel Standalone Profile Editor** from Cisco Software Download Center if you have not done already and create a profile with required settings for the Secure Client management VPN tunnel.

File Types	Details
AMP Enabler Service Profile	The profile is used for the Secure Client The AMP Enabler and this profile are deployed to endpoints from Firewall Threat Defense when a user connects to the VPN using remote access.
Feedback Profile	You can add a Customer Experience Feedback profile and select this type to receive information about the features and modules customers have enabled and use.
ISE Posture Profile	Choose if you are adding a profile file for the Secure Client ISE Posture module.
NAM Service Profile	Configure and add the NAM profile file using the Network Access Manager profile editor.
Network Visibility Service Profile	Profile file for Secure Client Network Visibility module. You can create the profile using the NVM profile editor.
Umbrella Roaming Security Profile	You must select this file type if you are deploying the Umbrella Roaming Security module using the .json file created using the profile editor.
Web Security Service Profile	Select this option when you add a profile file for the Web security module.
HostScan Posture Package	Select when you add a Secure Firewall Posture Package file. This file is used while configuring a Dynamic Access Policy (DAP) to collect information about the operating system, anti-virus, anti-spyware, and firewall software installed on the endpoints.
Secure Client External Browser Package	For selecting an external browser package file for SAML single sign-on web authentication. You can add the package file when a new version of the external package file is available. For more information, see Configure AAA Settings for Remote Access VPN.
Description	Add an optional description.

Related Topics

Cisco Secure Client Image
Group policy options for Secure Client, on page 133

History for object management

This reference provides a history of object management features, enhancements, and changes across releases, including details and supported platforms for each feature.

Feature	Minimum Firewall Management Center	Minimum Firewall Threat Defense	Details
Merged ACL and AV ACL	7.4.1	Any	New/modified screens: **Objects** > > **Object Management** > **RADIUS Server Group** > **Add RADIUS Server Group** > **Merge Downloadable ACL with Cisco AV Pair ACL** New CLI commands: • **sh run AAA-server AAA-server ISE-Server protocol RADIUS merge-dacl after-avpair** • **sh run AAA-server AAA-server ISE-Server protocol RADIUS merge-dacl before-avpair**
Secure Client Customization	Any	7.4	You can now customize Secure Client and deploy these customizations to the VPN headend. The threat defense distributes these customizations to the endpoint when an end user connects from the Secure Client.
IPv6 support for CRL and OCSP urls	Any	7.4	You can now configure IPv6 OCSP and CRL URLs.
Loopback and Management type interface group objects	Any	7.4	You can now create interface group objects that include only management-only interfaces or only loopback interfaces. You can then use these groups for management features such as DNS servers, HTTP access, or SSH. Loopback groups are supported for any feature that supports loopback interfaces. Note that DNS does not support management interfaces. New/Modified screens: **Objects** > **Object Management** > **Interface** > **Add** > **Interface Group**
Loopback interface support for AAA	Any	7.4	You can use a loopback interface group for configuring a RADIUS server. New/Modified screens: **Objects** > **Object Management** > **AAA Server** > **RADIUS Server Group**
Clone network and port objects	Any	7.4	You can now clone network and port objects. In the object manager (**Objects** > **Object Management**), click the new **Clone** icon next to a port or network object. You can then change the new object's properties and save it using a new name.

Feature	Minimum Firewall Management Center	Minimum Firewall Threat Defense	Details
DHCP IPv6 Pool	Any	7.3	The Firewall Threat Defense now supports a light DHCPv6 stateless server when using the DHCPv6 Prefix Delegation client. The Firewall Threat Defense provides other information such as the domain name to SLAAC clients when they send Information Request (IR) packets to the Firewall Threat Defense. The Firewall Threat Defense only accepts IR packets and does not assign addresses to the clients. New/Modified screens: • **Devices** > **Device Management** > **Interfaces** > **Add/Edit Interfaces** > **IPv6** > **DHCP** • **Objects** > **Object Management** > **DHCP IPv6 Pool** New/Modified commands: **show ipv6 DHCP**
BFD Template	Any	7.3	In the previous releases, BFD was configurable on threat defense only through FlexConfig. FlexConfig no longer supports BFD configuration. You can now configure BFD policies for threat defense in the management center UI. Hence, the BFD Template object was introduced. New/modified screens: • **Devices** > **Device Management** > **Routing** > **BFD**. • **Objects** > **Object Management** > **BFD Template**
New Applications tab for policy based routing	Any	7.1	A new tab for selecting the applications for configuring direct internet access policy (policy based routing) was introduced in the extended access list object. New/Modified Screens: New option to select applications when configuring **Object** > **Object Management** > **Access List** > **Extended** page. Supported platforms: Secure Firewall Management Center
New Extended Community List object and	Any	7.1	Extended community list object was introduced to be used in policy list and route map objects. The extended community list object is applicable only for importing or exporting routes in BGP route leaking support for virtual routers. New/Modified Screens: New object for configuring policy list and route maps **Object** > **Object Management** > **Community list** > **Extended Community** page. Supported platforms: Secure Firewall Management Center

Feature	Minimum Firewall Management Center	Minimum Firewall Threat Defense	Details
Enhancements to Policy List object and Route Map object	Any	7.1	Options to select the newly introduced Extended Community List objects in policy list and route maps. New/Modified Screens: New options for configuring policy list and route maps **Object** > **Object Management** > **Policy List** > **Community rule** tab, and **Object** > **Object Management** > **Route Map** > **BGP** > **Community List** tab. Supported platforms: Secure Firewall Management Center
Time-based ACL support for Snort 3	Any	7.0	Time-based rules in access control and prefilter policies are supported in Snort 3 as well. Supported platforms: Firewall Threat Defense
EST for certificate enrollment	Any	7.0	Support for Enrollment over Secure Transport for certificate enrollment was provided. New/Modified Screens: New enrollment options when configuring **Objects** > **PKI** > **Cert Enrollment** > **CA Information** tab. Supported platforms: Secure Firewall Management Center
Support for EdDSA certificate type	Any	7.0	A new certificate key type- EdDSA was added with a key size of 256. New/Modified Screens: New certificate key options when configuring **Objects** > **PKI** > **Cert Enrollmet** > **Key** tab. Supported platforms: Secure Firewall Management Center
Restrictions on ciphers and key sizes	Any	7.0	Certificates having SHA-1 with RSA Encryption signature algorithm, and RSA key sizes smaller than 2048 bits are not supported. To override these restrictions on existing certificates, you can enable the weak-crypto option on Firewall Threat Defense. However, you cannot generate RSA keys with sizes smaller than 2048 bits. New/Modified Screens: New toggle button when configuring **Devices** > **Certificates**. Supported platforms: Secure Firewall Management Center
Security Intelligence feed options	Any	6.7	New update frequency options (5 and 15 minutes) for custom Security Intelligence feeds. Update frequencies of less than 30 minutes require an MD5 URL, to prevent unnecessary downloads if the feed has not changed. New/Modified Screens: New frequency choices when configuring **Security Intelligence** > **Network Lists and Feeds**. Supported platforms: Secure Firewall Management Center

Feature	Minimum Firewall Management Center	Minimum Firewall Threat Defense	Details
Bulk upload of objects using a comma-separated-values (csv) file	Any	6.7	Objects can be imported from a comma-separated-values file. Up to 1000 objects can be imported in one attempt. New/Modified Screens: The following object types have a new **Import Object** option in the **Add [Object Type]** drop-down list. • **Distinguished Name** > **Individual Objects** • **Network Object** • **Port** • **URL** • **VLAN Tag** Supported platforms: Secure Firewall Management Center
See the policies in which interface objects are used	Any	6.6	See the policies in which interface objects are used. New/Modified Screens: The **Interface** object page in **Object** > **Object management** has a new **Find** button. Supported platforms: Secure Firewall Management Center
Time zone objects introduced	Any	6.6	You can assign time zones to Firewall Threat Defense devices, for use when applying time-based policies. New/Modified Screens: New **Time Zone Object** in **Object** > **Object management**. Supported platforms: Secure Firewall Management Center
Time-based objects can now be used in access control and prefilter policies	Any	6.6	Use time range objects in conjunction with new time zone objects for applying time-based rules in access control and prefilter policies. You can specify an absolute or recurring time or time range for a rule to be applied. The rule is applied based on the time zone of the device that processes the traffic.
View Object Details from prefilter rule page	Any	6.6	Feature introduced: Option to view details for an object or object group when viewing prefilter rules. New options: Right-clicking a value in any of the following columns in the prefilter rule list offers options to view object details: Source Networks, Destination Networks, Source Port, Destination Port, and VLAN Tag. Supported platforms: Secure Firewall Management Center