



Reusable Objects

The following topics describe how to manage reusable objects in the Firepower System:

- [Introduction to Reusable Objects, on page 1](#)
- [The Object Manager, on page 3](#)
- [Network Objects, on page 9](#)
- [Port Objects, on page 10](#)
- [Application Filters, on page 11](#)
- [VLAN Tag Objects, on page 12](#)
- [URL Objects, on page 12](#)
- [Geolocation Objects, on page 14](#)
- [Security Zones, on page 14](#)
- [Variable Sets, on page 16](#)
- [Security Intelligence Lists and Feeds, on page 31](#)
- [Sinkhole Objects, on page 42](#)
- [File Lists, on page 42](#)
- [Cipher Suite Lists, on page 47](#)
- [Distinguished Name Objects, on page 48](#)
- [PKI Objects, on page 51](#)

Introduction to Reusable Objects

For increased flexibility and web interface ease-of-use, the Firepower System uses named *objects*, which are reusable configurations that associate a name with a value. When you want to use that value, use the named object instead. The system supports object use in various places in the web interface, including many policies and rules, event searches, reports, dashboards, and so on. The system provides many predefined objects that represent frequently used configurations.

Use the object manager to create and manage objects. Many configurations that use objects also allow you to create objects on the fly, as needed. You can also use the object manager to:

- View the policies, settings, and other objects where a network, port, VLAN, or URL object is used; see [Viewing Objects and Their Usage, on page 4](#).
- Group objects to reference multiple objects with a single configuration; see [Object Groups, on page 5](#).
- Override object values for selected devices or, in a multidomain deployment, selected domains; see [Object Overrides, on page 6](#).

After you edit 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 An object is configured on a managed device if, and only if, the object is used in a policy that is assigned to that device. 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

The following table lists the objects you can create in the Firepower 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
Security Zone	no	no
Application Filter	no	no
VLAN Tag	yes	yes
URL	yes	yes
Geolocation	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): <ul style="list-style-type: none"> • Internal and Trusted CA • Internal and External Certs 	yes	no

Objects and Multitenancy

In a multidomain deployment, you can create objects in Global and descendant domains. The system displays objects created in the current domain, which you can edit. It also displays objects created in ancestor domains, which you cannot edit, with the exception of security zones.



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

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.

The Object Manager

You can use the object manager to create and manage 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.

By default, the page lists objects and groups alphabetically by name. You can filter the objects on the page by name or value.

Editing Objects

In a multidomain deployment, the system displays objects created in the current domain, which you can edit. It also displays objects created in ancestor domains, which in most cases you cannot edit. To view and edit objects in a descendant domain, switch to that domain.

Procedure

- Step 1** Choose **Objects > Object Management**.
- Step 2** Choose an object type from the list; see [Introduction to Reusable Objects, on page 1](#).
- 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 [Managing Variables, on page 28](#).
- 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 [Allowing Object Overrides, on page 8](#). 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 [Adding Object Overrides, on page 8](#).
- 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](#).

Viewing Objects and Their Usage

You can view usage details of objects on the Object Management page. The network, port, VLAN, and URL object types are the only object types that provide this functionality.



Note In a multidomain deployment, you can view objects from any other domain. However, to find usage of objects in a descendant domain, switch to that domain.

Procedure

- Step 1** Choose **Objects > Object Management**.
- Step 2** Choose one of the following object types:
- Network
 - Port
 - VLAN Tag
 - URL
- Step 3** Click **Find Usage** (🔍) 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.
-

Filtering Objects or Object Groups

In a multidomain deployment, the system displays objects created in the current and ancestor domains, which you can filter.

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 the following 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.
-

Object Groups

Grouping objects allows you to reference multiple objects with a single configuration. 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.

You can group network, port, VLAN tag, URL, and PKI objects.

Objects and object groups of the same type cannot have the same name. In a multidomain deployment, the names of object groups must be unique within the domain hierarchy. Note that the system may identify a conflict with the name of an object group you cannot view in your current domain.

When you edit an object group used in a policy (for example, a network object group used in an access control policy), you must re-deploy the changed configuration for your changes to take effect.

Deleting a group does not delete the objects in the group, just 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 you are using in a VLAN condition in a saved access control policy.

Grouping Reusable Objects

In a multidomain deployment, the system displays objects created in the current domain, which you can edit. It also displays objects created in ancestor domains, which in most cases you cannot edit. To view and edit objects in a descendant domain, switch to that domain.

You can group objects in the current domain with objects inherited from ancestor domains.

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 [Allowing Object Overrides, on page 8](#).

Step 8 Click **Save**.

What to do next

- If an active policy references your object group, deploy configuration changes; see [Deploy Configuration Changes](#).

Object Overrides

An object override allows you to define an alternate value for an object, which the system uses for the devices you specify.

You can create an object whose definition works for most devices, and then use overrides to specify modifications to the object for the few devices that need different definitions. You can also create an object that needs to be overridden for all devices, but its use allows you to create a single policy for all devices. Object overrides allow you to create a smaller set of shared policies for use across devices without giving up the ability to alter policies when needed for individual devices.

For example, you might want to deny ICMP traffic to the different departments in your company, each of which is connected to a different network. You can do this by defining an access control policy with a rule that includes a network object called Departmental Network. By allowing overrides for this object, you can then create overrides on each relevant device that specifies the actual network where that device is connected.

In a multidomain deployment, you can define a default value for an object in an ancestor domain and allow administrators in descendant domains to add override values for that object. For example, a managed security service provider (MSSP) might use a single Firepower Management Center to manage network security for multiple customers. Administrators at the MSSP can define an object in the Global domain for use in all customers' deployments. Administrators for each customer can log into descendant domains to override that object for their organizations. These local administrators cannot view or affect the override values of other customers of the MSSP.

You can target an object override to a specific domain. In this case, the system uses the object override value for all devices in the targeted domain unless you override it at the device level.

From the object manager, you can choose an object that can be overridden and define a list of device-level or domain-level overrides for that object.

You can use object overrides with the following object types only:


- Network
- Port
- VLAN tag
- URL

If you can override an object, the **Override** column appears for the object type in the object manager. Possible values for this column include:

- Green checkmark — indicates that you can create overrides for the object and no overrides have been added yet
- Red X — indicates that you cannot create overrides for the object
- Number — represents a count of the overrides that have been added to that object (for example, "2" indicates two overrides have been added)

Managing Object Overrides

Procedure

- Step 1** Choose **Objects > Object Management**.
- Step 2** Choose from the list of object types; see [Introduction to Reusable Objects, on page 1](#).
- 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 [Adding Object Overrides, on page 8](#).
 - Allow—Allow object overrides; see [Allowing Object Overrides, on page 8](#).
 - Delete—In the object editor, click **Delete** (🗑) next to the override you want to remove.
 - Edit—Edit object overrides; see [Editing Object Overrides, on page 9](#).
-

Allowing Object Overrides

Procedure

- Step 1** In the object editor, check the **Allow Overrides** check box.
Step 2 Click **Save**.
-

What to do next

Add object override values; see [Adding Object Overrides, on page 8](#).

Adding Object Overrides

Before you begin

Allow object overrides; see [Allowing Object Overrides, on page 8](#).

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](#).

Editing Object Overrides

You can modify the description and the value of an existing override, but you cannot modify the existing target list. Instead, you must add a new override with new targets, which replaces 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](#).

Network Objects

A network object represents one or more IP addresses that you can specify either individually or as address blocks. You can use network objects and groups in various places in the system's web interface, including access control policies, network variables, identity rules, network discovery rules, event searches, reports, and so on.

Creating Network Objects

Procedure

- Step 1** Choose **Objects > Object Management**.
- Step 2** Choose **Network** from the list of object types.
- Step 3** Choose **Add Object** from the **Add Network** drop-down menu.
- Step 4** Enter a **Name**.

In a multidomain deployment, 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.

- Step 5** Optionally, enter a **Description**.
- Step 6** In the **Network** field, enter an IP address or address block to add to the object.
- Step 7** Manage overrides for the object:
- If you want to allow overrides for this object, check the **Allow Overrides** check box; see [Allowing Object Overrides, on page 8](#).
 - If you want to add override values to this object, expand the Override section and click **Add**; see [Adding Object Overrides, on page 8](#).
- Step 8** Click **Save**.
-

What to do next

- If an active policy references your object, deploy configuration changes; see [Deploy Configuration Changes](#).

Port Objects

Port objects represent different protocols in slightly different ways:

TCP and UDP

A port object represents the transport layer protocol, with the protocol number in parentheses, plus an optional associated port or port range. For example: `TCP (6) /22`.

ICMP and ICMPv6 (IPv6-ICMP)

A port object represents the Internet layer protocol plus an optional type and code. For example: `ICMP (1) : 3:3`.

You can restrict an ICMP or IPV6-ICMP port object by type and, if applicable, code. For more information on ICMP types and codes, see:

- <http://www.iana.org/assignments/icmp-parameters/icmp-parameters.xml>
- <http://www.iana.org/assignments/icmpv6-parameters/icmpv6-parameters.xml>

Other

A port object can represent other protocols that do not use ports.

The Firepower System provides default port objects for well-known ports. You cannot modify or delete these default objects. You can create custom port objects in addition to the default objects.

You can use port objects and groups in various places in the system's web interface, including access control policies, identity rules, network discovery rules, port variables, and event searches. For example, if your organization uses a custom client that uses a specific range of ports and causes the system to generate excessive and misleading events, you can configure your network discovery policy to exclude monitoring those ports.

When using port objects, observe the following guidelines:

- You cannot add any protocol other than TCP or UDP for source port conditions in access control rules. Also, you cannot mix transport protocols when setting both source and destination port conditions in a rule.

- If you add an unsupported protocol to a port object group used in a source port condition, the rule where it is used does not take effect on the managed device when the configuration is deployed.
- If you create a port object containing both TCP and UDP ports, then add it as a source port condition in a rule, you cannot add a destination port, and vice versa.

Creating Port Objects

Procedure

- Step 1** Choose **Objects > Object Management**.
- Step 2** Choose **Port** from the list of object types.
- Step 3** Choose **Add Object** from the **Add Port** drop-down list.
- 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 [Allowing Object Overrides, on page 8](#).
 - If you want to add override values to this object, expand the Override section and click **Add**; see [Adding Object Overrides, on page 8](#).
- Step 8** Click **Save**.
-

What to do next

- If an active policy references your object, deploy configuration changes; see [Deploy Configuration Changes](#).

Application Filters

System-provided application filters help you perform application control by organizing applications according to basic characteristics: type, risk, business relevance, category, and tags. In the object manager, you can create and manage reusable user-defined application filters based on combinations of the system-provided filters, or on custom combinations of applications. For detailed information, see [Application Conditions \(Application Control\)](#).

VLAN Tag Objects

Each VLAN tag object you configure represents a VLAN tag or range of tags.

You can group VLAN tag objects. Groups represent multiple objects; using a range of VLAN tags in a single object is not considered a group in this sense.

You can use VLAN tag objects and groups in various places in the system's web interface, including rules and event searches. For example, you could write an access control rule that applies only to a specific VLAN.

Creating VLAN Tag Objects

Procedure

- Step 1** Choose **Objects > Object Management**.
 - Step 2** Choose **VLAN Tag** from the list of object types.
 - Step 3** Choose **Add Object** from the **Add VLAN Tag** drop-down list.
 - Step 4** Enter a **Name**.
 - Step 5** Enter a **Description**.
 - Step 6** Enter a value in the **VLAN Tag** field. Use a hyphen to specify a range of VLAN tags.
 - Step 7** Manage overrides for the object:
 - If you want to allow overrides for this object, check the **Allow Overrides** check box; see [Allowing Object Overrides, on page 8](#).
 - If you want to add override values to this object, expand the Override section and click **Add**; see [Adding Object Overrides, on page 8](#).
 - Step 8** Click **Save**.
-

What to do next

- If an active policy references your object, deploy configuration changes; see [Deploy Configuration Changes](#).

URL Objects



Important For best practices for using this and similar options in Security Intelligence configurations and for URL rules in access control policies, see [Manual URL Filtering Options](#).

A URL object defines a single URL or IP address, whereas a URL group object can define more than one URL or address. You can use URL objects and groups in various places in the system's web interface, including access control policies and event searches.

The system makes a simple substring match on any URL that you enter, which may not necessarily be what you expect. See matching example URLs at [Manual URL Filtering Options](#).

When creating URL objects, especially if you do not configure SSL inspection to decrypt or block encrypted traffic, keep the following points in mind:

- If you plan to use a URL object to match HTTPS traffic in an access control rule, create the object using the subject common name in the public key certificate used to encrypt the traffic. Also, the system disregards subdomains within the subject common name, so do not include subdomain information. For example, use `example.com` rather than `www.example.com`.
- When matching web traffic using access control rules with URL conditions, the system disregards the encryption protocol (HTTP vs HTTPS). In other words, if you block a website, both HTTP and HTTPS traffic to that website is blocked, unless you use an application condition to refine the rule. When creating a URL object, you do not need to specify the protocol when creating an object. For example, use `example.com` rather than `http://example.com/`.

Creating URL Objects

Procedure

- Step 1** Choose **Objects > Object Management**.
- Step 2** Choose **URL** from the list of object types.
- Step 3** Choose **Add Object** from the **Add URL** drop-down list.
- Step 4** Enter a **Name**.

In a multidomain deployment, 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.

- Step 5** Optionally, enter a **Description**.
- Step 6** Enter the **URL** or IP address.
- Step 7** Manage overrides for the object:
- If you want to allow overrides for this object, check the **Allow Overrides** check box; see [Allowing Object Overrides, on page 8](#).
 - If you want to add override values to this object, expand the Override section and click **Add**; see [Adding Object Overrides, on page 8](#).
- Step 8** Click **Save**.
-

What to do next

- If an active policy references your object, deploy configuration changes; see [Deploy Configuration Changes](#).

Geolocation Objects

Each geolocation object you configure represents one or more countries or continents that the system has identified as the source or destination of traffic on your monitored network. You can use geolocation objects in various places in the system's web interface, including access control policies, SSL policies, and event searches. For example, you could write an access control rule that blocks traffic to or from certain countries.

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

Creating Geolocation Objects

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**.

In a multidomain deployment, 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.

Step 5 Check the check boxes for the countries and continents you want to include in your geolocation object. Checking a continent chooses all countries within that continent, as well as any countries that GeoDB updates may add under that continent in the future. Unchecking any country under a continent unchecks the continent. You can choose 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](#).

Security Zones

Security zones segment your network to help you manage and classify traffic flow. A security zone simply groups interfaces. These groups may span multiple devices; you can also configure multiple zones on a single device.

All interfaces in a security zone must be of the same type: all inline, passive, switched, routed, or ASA FirePOWER. After you create a security zone, you cannot change the type of interfaces it contains. An interface can belong to only one zone.

The Security Zones page of the object manager lists the zones configured on your managed devices. The page also displays the type of interfaces in each zone, and you can expand each zone to view which interfaces on which devices belong to each zone.



Note Create inline sets before you add security zones for the interfaces in the inline set; otherwise security zones are removed and you must add them again.

Model-Specific Notes and Warnings

During initial configuration of a 7000 or 8000 Series device, the system creates security zones based on the detection mode you selected for the device. For example, the system creates a Passive zone in passive deployments, while in inline deployments the system creates External and Internal zones. When you register the device to the Firepower Management Center, those security zones are added to the FMC.

Zones and Multitenancy

In a multidomain deployment, you can create security zones at any level. A zone created in an ancestor domain can contain interfaces that reside on devices in different domains. In this situation, subdomain users viewing the ancestor zone configuration in the object manager can see only the interfaces in their domain.

Unless restricted by role, subdomain users can view **and** edit zones created in ancestor domains. Subdomain users can add and delete interfaces from these zones. They cannot, however, delete or rename the zones. You can neither view nor edit zones created in descendant domains.

Creating Security Zone Objects



Tip You can create empty security zones and add interfaces to them later. To add an interface, the interface must have a name. You can also create security zones while configuring interfaces in **Devices > Device Management**.

Before you begin

- Understand the usage requirements and restrictions for each type of security zone. See [Security Zones, on page 14](#).

Procedure

- Step 1** Choose **Objects > Object Management**.
- Step 2** Choose **Security Zones** from the list of object types.
- Step 3** Click **Add Security Zone**.
- Step 4** Enter a **Name**.
- Step 5** Choose an **Interface Type**.
- Step 6** From the **Device > Interfaces** drop-down list, choose a device that contains interfaces you want to add.
- Step 7** Choose one or more interfaces.

Step 8 Click **Add** to add the interfaces you chose, grouped by device.

Step 9 Click **Save**.

What to do next

- If an active policy references your object, deploy configuration changes; see [Deploy Configuration Changes](#).

Variable Sets

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 profiles, 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 Firepower 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 Firepower 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 Cisco Talos Intelligence Group (Talos) 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.

In a multidomain deployment, the system generates a default variable set for each subdomain.



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

[Managing Variables](#), on page 28

[Managing Variable Sets](#), on page 26

Variable Sets in Intrusion Policies

By default, the Firepower 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

Variables belong to one of the following categories:

Default Variables

Variables provided by the Firepower 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

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. Because variable values in the default set determine the default values of variables in custom sets, customizing a default variable in the default set modifies 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 the following 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.

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.

Advanced Variables

Variables provided by the Firepower System under specific conditions. These variables have a very limited deployment.

Predefined Default Variables

By default, the Firepower System provides a single default variable set, which is comprised of predefined default variables. The Cisco Talos Intelligence Group (Talos) uses rule updates to provide new and updated intrusion rules and other intrusion policy elements, including default variables.

Because many intrusion rules provided by the system use predefined default variables, you should set appropriate values for these variables. Depending on how you use variable sets to identify traffic on your network, you can modify the values for these default variables in any or all variable sets.



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.

The following 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 1: System-Provided Variables

Variable Name	Description	Modify?
<code>\$AIM_SERVERS</code>	Defines known AOL Instant Messenger (AIM) servers, and is used in chat-based rules and rules that look for AIM exploits.	Not required.
<code>\$DNS_SERVERS</code>	Defines Domain Name Service (DNS) servers. If you create a rule that affects DNS servers specifically, you can use the <code>\$DNS_SERVERS</code> variable as a destination or source IP address.	Not required in current rule set.

Variable Name	Description	Modify?
\$EXTERNAL_NET	Defines the network that the Firepower System views as the unprotected network, and is used in many rules to define the external network.	Yes, you should adequately define \$HOME_NET and then exclude \$HOME_NET as the value for \$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 servers use ports other than the default ports (you can view the default ports in 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 addresses for 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 servers use ports other than the default ports (you can view the default ports in the web interface).
\$HTTP_SERVERS	Defines the web servers on your network. Used in web server exploit rules.	Yes, if you run HTTP servers.
\$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 Oracle servers.
\$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.
\$SIP_SERVERS	Defines SIP servers on your network, and is used in rules that address SIP-targeted exploits.	Yes, if you run SIP servers, you should adequately define \$HOME_NET and then include \$HOME_NET as the value for \$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 servers.
\$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 servers.
\$SNORT_BPF	Identifies a legacy advanced variable that appears only when it existed on your system in a Firepower System software release before Version 5.3.0 that you subsequently upgraded to Version 5.3.0 or greater.	No, you can only view or delete this variable. You cannot edit it or recover it after deleting it.

Variable Name	Description	Modify?
<code>\$SQL_SERVERS</code>	Defines database servers on your network, and is used in rules that address database-targeted exploits.	Yes, if you run SQL servers.
<code>\$SSH_PORTS</code>	Defines the ports of SSH servers on your network, and is used for SSH server exploit rules.	Yes, if your SSH servers use ports other than the default port (you can view the default ports in the web interface).
<code>\$SSH_SERVERS</code>	Defines SSH servers on your network, and is used in rules that address SSH-targeted exploits.	Yes, if you run SSH servers, you should adequately define <code>\$HOME_NET</code> and then include <code>\$HOME_NET</code> as the value for <code>\$SSH_SERVERS</code> .
<code>\$TELNET_SERVERS</code>	Defines known Telnet servers on your network, and is used in rules that address Telnet server-targeted exploits.	Yes, if you run Telnet servers.
<code>\$USER_CONF</code>	Provides a general tool that allows you to configure one or more features not otherwise available via the web interface. Conflicting or duplicate <code>\$USER_CONF</code> configurations will halt the system.	No, only as instructed in a feature description or with the guidance of Support.

Network Variables

Network variables represent IP addresses you can use in intrusion rules that you enable in an intrusion policy and in intrusion policy rule suppressions, dynamic rule states, and adaptive profiles. Network variables differ from network objects and network object groups in that network variables are specific to intrusion policies and intrusion rules, whereas you can use network objects and groups to represent IP addresses in various places in the system's web interface, including access control policies, network variables, intrusion rules, network discovery rules, event searches, reports, and so on.

You can use network variables in the following configurations to specify the IP addresses of hosts on your network:

- intrusion rules—Intrusion rule **Source IPs** and **Destination IPs** header fields allow you to restrict packet inspection to the packets originating from or destined to specific IP addresses.
- suppressions—The **Network** field in source or destination intrusion rule suppressions allows you to suppress intrusion event notifications when a specific IP address or range of IP addresses triggers an intrusion rule or preprocessor.
- dynamic rule states—The **Network** field in source or destination dynamic rule states allows you to detect when too many matches for an intrusion rule or preprocessor rule occur in a given time period.
- adaptive profiles—The adaptive profiles **Networks** field identifies hosts where you want to improve reassembly of packet fragments and TCP streams in passive deployments.

When you use variables in the fields identified in this section, the variable set you link to an intrusion policy determines the variable values in the network traffic handled by an access control policy that uses the intrusion policy.

You can add any combination of the following network configurations to a variable:

- any combination of network variables, network objects, and network object groups that you select from the list of available networks
- individual network objects that you add from the New Variable or Edit Variable page, and can then add to your variable and to other existing and future variables
- literal, single IP addresses or address blocks

You can list multiple literal IP addresses and address blocks by adding each individually. You can list IPv4 and IPv6 addresses and address blocks alone or in any combination. When specifying IPv6 addresses, you can use any addressing convention defined in RFC 4291.

The default value for included networks in any variable you add is the word `any`, which indicates any IPv4 or IPv6 address. The default value for excluded networks is `none`, which indicates no network. You can also specify the address `::` in a literal value to indicate any IPv6 address in the list of included networks, or no IPv6 addresses in the list of exclusions.

Adding networks to the excluded list negates the specified addresses and address blocks. That is, you can match any IP address with the exception of the excluded IP address or address blocks.

For example, excluding the literal address `192.168.1.1` specifies any IP address other than `192.168.1.1`, and excluding `2001:db8:ca2e::fa4c` specifies any IP address other than `2001:db8:ca2e::fa4c`.

You can exclude any combination of networks using literal or available networks. For example, excluding the literal values `192.168.1.1` and `192.168.1.5` *includes* any IP address other than `192.168.1.1` or `192.168.1.5`. That is, the system interprets this as “**not** `192.168.1.1` **and not** `192.168.1.5`,” which matches any IP address other than those listed between brackets.

Note the following points when adding or editing network variables:

- You cannot logically exclude the value `any` which, if excluded, would indicate no address. For example, you cannot add a variable with the value `any` to the list of excluded networks.
- Network variables identify traffic for the specified intrusion rule and intrusion policy features. Note that preprocessor rules can trigger events regardless of the hosts defined by network variables used in intrusion rules.
- Excluded values must resolve to a subset of included values. For example, you cannot include the address block `192.168.5.0/24` and exclude `192.168.6.0/24`.

Port Variables

Port variables represent TCP and UDP ports you can use in the **Source Port** and **Destination Port** header fields in intrusion rules that you enable in an intrusion policy. Port variables differ from port objects and port object groups in that port variables are specific to intrusion rules. You can create port objects for protocols other than TCP and UDP, and you can use port objects in various places in the system’s web interface, including port variables, access control policies, network discovery rules, and event searches.

You can use port variables in the intrusion rule **Source Port** and **Destination Port** header fields to restrict packet inspection to packets originating from or destined to specific TCP or UDP ports.

When you use variables in these fields, the variable set you link to the intrusion policy associated with an access control rule or policy determines the values for these variables in the network traffic where you deploy the access control policy.

You can add any combination of the following port configurations to a variable:

- any combination of port variables and port objects that you select from the list of available ports

Note that the list of available ports does not display port object groups, and you cannot add these to variables.

- individual port objects that you add from the New Variable or Edit Variable page, and can then add to your variable and to other existing and future variables

Only TCP and UDP ports, including the value `any` for either type, are valid variable values. If you use the new or edit variables page to add a valid port object that is not a valid variable value, the object is added to the system but is not displayed in the list of available objects. When you use the object manager to edit a port object that is used in a variable, you can only change its value to a valid variable value.

- single, literal port values and port ranges

You must separate port ranges with a dash (-). Port ranges indicated with a colon (:) are supported for backward compatibility, but you cannot use a colon in port variables that you create.

You can list multiple literal port values and ranges by adding each individually in any combination.

Note the following points when adding or editing port variables:

- The default value for included ports in any variable you add is the word `any`, which indicates any port or port range. The default value for excluded ports is `none`, which indicates no ports.



Tip To create a variable with the value `any`, name and save the variable without adding a specific value.

- You cannot logically exclude the value `any` which, if excluded, would indicate no ports. For example, you cannot save a variable set when you add a variable with the value `any` to the list of excluded ports.
- Adding ports to the excluded list negates the specified ports and port ranges. That is, you can match any port with the exception of the excluded ports or port ranges.
- Excluded values must resolve to a subset of included values. For example, you cannot include the port range 10-50 and exclude port 60.

Advanced Variables

Advanced variables allow you to configure features that you cannot otherwise configure via the web interface. The Firepower System currently provides only one advanced variable, the `USER_CONF` variable.

USER_CONF

`USER_CONF` provides a general tool that allows you to configure 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 total characters on a single line; the line wraps automatically. You can include any number of valid instructions or lines until you reach the 8192 maximum character length for a variable or a physical limit such as disk space. Use the backslash (\) line continuation character after any complete argument in a command directive.

Resetting USER_CONF empties it.

Variable Reset

You can reset a variable to its default value on the variable set new or edit variables page. The following table summarizes the basic principles of resetting variables.

Table 2: 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)

Resetting a variable in a custom set simply resets it to the current value for that variable in the default set.

Conversely, resetting or modifying the value of a variable in the default set always updates the default value of that variable in all custom sets. When the reset icon is grayed out, indicating that you cannot reset the variable, this means that the variable has no customized value in that set. Unless you have customized the value for a variable in a custom set, a change to the variable in the default set updates the value used in any intrusion policy where you have linked the variable set.



Note It is good practice when you modify a variable in the default set to assess how the change affects any intrusion policy that uses the variable in a linked custom set, especially when 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. When the customized value and the reset value are the same, this indicates one of the following:

- 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

Adding Variables to Sets

Adding a variable to a variable set adds it to all other sets. When you add a variable from a custom set, you must choose whether to use the configured value as the customized value in the default set:

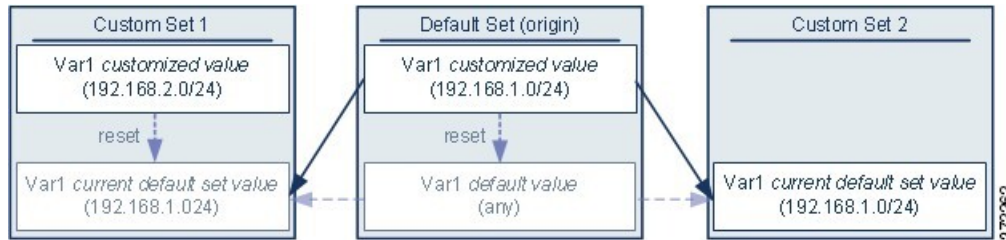
- **If you use the configured value** (for example, 192.168.0.0/16), the variable is added to the default set using the configured value as a customized value with a default value of `any`. Because the current value in the default set determines the default value in other sets, the initial, default value in other custom sets is the configured value (which in the example is 192.168.0.0/16).

Example: Adding User-Defined Variables to Default Sets

- **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`.

Example: Adding User-Defined Variables to Default Sets

The following diagram illustrates set interactions when you add the user-defined variable `var1` to the default set with the value `192.168.1.0/24`.



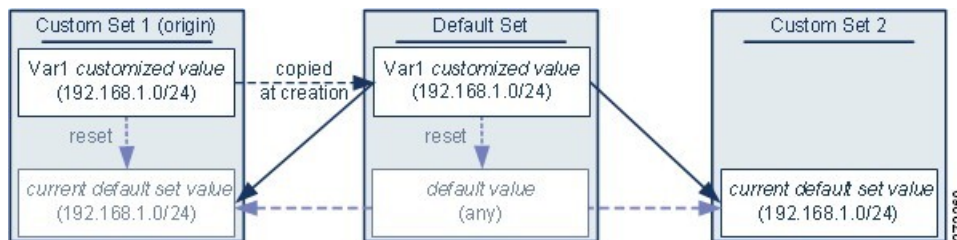
You can customize the value of `var1` in any set. In Custom Set 2 where `var1` has not been customized, its value is `192.168.1.0/24`. In Custom Set 1 the customized value `192.168.2.0/24` of `var1` overrides the default value. Resetting a user-defined variable in the default set resets its default value to `any` in all sets.

It is important to note in this example that, if you do not update `var1` in Custom Set 2, further customizing or resetting `var1` in the default set consequently updates the current, default value of `var1` in Custom Set 2, thereby affecting any intrusion policy linked to the variable set.

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.

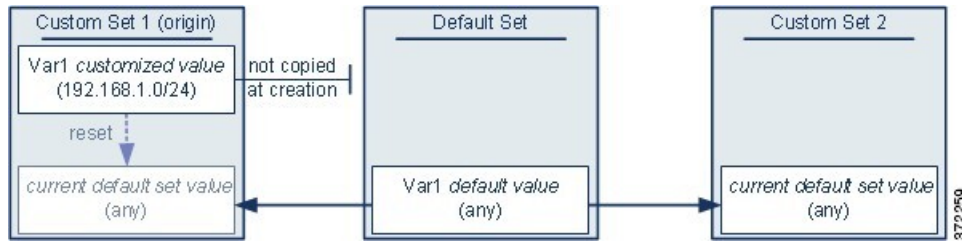
Example: Adding User-Defined Variables to Custom Sets

The next two examples illustrate variable set interactions when you add a user-defined variable to a custom set. When you save the new variable, you are prompted whether to use the configured value as the default value for other sets. In the following example, you elect **to use** the configured value.



Note that, except for the origin of `var1` from Custom Set 1, this example is identical to the example above where you added `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. As with the previous example, keep in mind that further customizing or resetting `var1` in the default set consequently updates the current, default value of `var1` in Custom Set 2, thereby affecting any intrusion policy linked to the variable set.

In the next example, you add `var1` with the value `192.168.1.0/24` to Custom Set 1 as in the previous example, but you elect **not to use** the configured value of `var1` as the default value in other sets.



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 thus inadvertently changing the current value in a set such as Custom Set 2 where you have not customized `Var1`.

Nesting Variables

You can nest variables so long as the nesting is not circular. Nested, negated variables are not supported.

Valid Nested Variables

In this example, `SMTP_SERVERS`, `HTTP_SERVERS`, and `OTHER_SERVERS` are valid nested variables.

Variable	Type	Included Networks	Excluded Networks
<code>SMTP_SERVERS</code>	customized default	10.1.1.1	—
<code>HTTP_SERVERS</code>	customized default	10.1.1.2	—
<code>OTHER_SERVERS</code>	user-defined	10.2.2.0/24	—
<code>HOME_NET</code>	customized default	10.1.1.0/24 <code>OTHER_SERVERS</code>	<code>SMTP_SERVERS</code> <code>HTTP_SERVERS</code>

An 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
<code>SMTP_SERVERS</code>	customized default	10.1.1.1	—
<code>HTTP_SERVERS</code>	customized default	10.1.1.2	—
<code>OTHER_SERVERS</code>	user-defined	10.2.2.0/24 <code>HOME_NET</code>	—

Variable	Type	Included Networks	Excluded Networks
HOME_NET	customized default	10.1.1.0/24 OTHER_SERVERS	SMTP_SERVERS HTTP_SERVERS

An Unsupported Nested, Negated Variable

Because nested, negated variables are not supported, you cannot use the variable NONCORE_NET as shown in this example to represent IP addresses that are outside of 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 an 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 as shown in this example.

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

Managing Variable Sets

To use variable sets, you must have the Threat license (for FTD devices) or the Protection license (all other device types).

In a multidomain deployment, the system displays objects created in the current domain, which you can edit. It also displays objects created in ancestor domains, which in most cases you cannot edit. To view and edit objects in a descendant domain, switch to that domain.


Procedure

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

Step 2 Choose **Variable Set** from the list of object types.


Step 3 Manage your variable sets:

- **Add** — If you want to add a custom variable set, click **Add Variable Set**; see [Creating Variable Sets, on page 27](#).

- **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 [Editing Objects, on page 3](#).

- **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 [Managing Variables, on page 28](#).

Creating Variable Sets

Procedure

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

Step 2 Choose **Variable Set** from the list of object types.

Step 3 Click **Add Variable Set**.

Step 4 Enter a **Name**.

In a multidomain deployment, 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.

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

Step 6 Manage the variables in the set; see [Managing Variables, on page 28](#).

Step 7 Click **Save**.

What to do next

- If an active policy references your object, deploy configuration changes; see [Deploy Configuration Changes](#).

Managing Variables


You must have the Threat license (for FTD devices) or the Protection license (all other device types).


In a multidomain deployment, the system displays objects created in the current domain, which you can edit. It also displays objects created in ancestor domains, which in most cases you cannot edit. To view and edit objects in a descendant domain, switch to that domain.

Procedure


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

Step 2 Choose **Variable Set** from the list of object types.


Step 3 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 4 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 [Adding Variables, on page 29](#).
- **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 the following:

- 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 [Editing Variables, on page 30](#).
- **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 the following 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 5 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.

Because the current value in the default set determines the default value in all other sets, modifying or resetting a variable in the default set changes the current value 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](#).

Adding Variables

You must have the Threat license (for FTD devices) or the Protection license (all other device types).

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 Specify values for 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 choose one or more items and then drag and drop, or click **Include** or **Exclude**.

Tip If addresses or ports in the included and excluded lists for a network or port variable overlap, 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 5 Click **Save** to save the variable. If you are adding a new variable from a custom set, you have the following options:

- Click **Yes** to add the variable using the configured value as the customized value in the default set and, consequently, the default value in other custom sets.
- Click **No** to add the variable as the default value of `any` in the default set and, consequently, in other custom sets.

Step 6 Click **Save** to save the variable set. 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](#).

Editing Variables


You must have the Threat license (for FTD devices) or the Protection license (all other device types).


In a multidomain deployment, the system displays objects created in the current domain, which you can edit. It also displays objects created in ancestor domains, which in most cases you cannot edit. To view and edit objects in a descendant domain, switch to that domain.

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 in the included and excluded lists for a network or port variable overlap, 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](#).

Security Intelligence Lists and Feeds

Security Intelligence functionality requires the Threat license (for FTD devices) or the Protection license (all other device types).

Security Intelligence *lists* and *feeds* are collections of IP addresses, domain names, and URLs that you can use 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 the following 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.

Predefined Lists: Global Block Lists and Global Do Not Block Lists

The system ships with 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 32](#).

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 Firepower Management Center appliances.

See [Custom Security Intelligence Feeds, on page 38](#).

Custom Lists

Custom lists can augment and fine-tune feeds and the Global lists.

See [Custom Security Intelligence Lists, on page 40](#).

Where Security Intelligence Lists and Feeds Are Used

- 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 rules, whose analysis and traffic handling phases occur after Security Intelligence.

How to Modify Security Intelligence Objects

To add or delete entries on a Block list, Do Not Block list, feed, or sinkhole object:

Object Type	Edit Capabilities	Requires Redeploy After Edit?
Custom Block and Do Not Block lists	Upload new and replacement lists using the object manager.	Yes
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

Firepower Management Center ships with empty Global Block and Do-Not-Block lists to which you can instantly add URLs, domains, and IP addresses from events on your network at any time. These lists allow you to use Security Intelligence to always block particular connections, or to exempt particular connections from blocking by Security Intelligence, allowing them to be evaluated by other threat detection processes that you have configured.

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.

By default, Access control and DNS policies use these Global lists, which apply to all security zones. You can opt not to use these lists on a per-policy basis.



Note These options apply to Security Intelligence only. Security Intelligence cannot block traffic that has already been fastpathed. Similarly, adding an item to a Security Intelligence Do Not Block list does not automatically trust or fastpath matching traffic. For more information, see [About Security Intelligence](#).

In a multidomain deployment, you can choose the Firepower System domains where you want to enforce blocking, or exempting from Security Intelligence blocking, by adding items to Domain lists as well as the Global lists; see [Security Intelligence Lists and Multitenancy, on page 33](#).

Security Intelligence Lists and Multitenancy

In a multidomain deployment, the Global domain owns the Global Block lists and Do Not Block lists. Only Global administrators can add to or remove items from the Global lists. So that subdomain users can add networks, domain names, and URLs to Block and Do Not Block lists, multitenancy adds:

- Domain lists—Block or Do Not Block lists whose contents apply to a particular subdomain only. The Global lists are Domain lists for the Global domain.
- Descendant Domain lists—Block or Do Not Block lists that aggregate the Domain lists of the current domain's descendants.

Domain Lists

In addition to being able to access (but not edit) the Global lists, each subdomain has its own named lists, the contents of which apply only to that subdomain. For example, a subdomain named Company A owns:

- 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 choose to add the same IP address to the Block list in the Global domain and Company A's domain, but not add it to the Block list in Company B's domain. This action would add the same IP address to:

- Global Block list (where it can be removed only by Global administrators)
- Domain Block list - Company A (where it can be removed only by Company A administrators)

The system builds a separate network map for each leaf domain. In a multidomain deployment, using literal IP addresses to constrain this configuration can have unexpected results.

Descendant Domain Lists

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 still allowing subdomain users to add items to a Block or Do Not Block list in their own deployment.

For example, the Global domain has the following 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

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 32](#).

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
- A combination of roles: 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

If appropriate, verify that these lists are used in the policies in which you expect them to be used.

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:

Target Item	Context Menu Option	Affected Global Lists
An IP address	Blacklist Now	Global Block List
	Whitelist Now	Global Whitelist

Target Item	Context Menu Option	Affected Global Lists
A URL	Blacklist HTTP/S Connections to URL Now Whitelist HTTP/S Connections to URL Now	Global Block List for URL Global Whitelist for URL
An entire domain	Blacklist HTTP/S Connections to Domain Now Whitelist HTTP/S Connections to Domain Now	Global Block List for URL Global Whitelist for URL
DNS requests for an entire domain	Blacklist DNS Requests to Domain Now Whitelist DNS Requests to Domain Now	Global Block List for DNS Global Whitelist 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 35](#).

Delete Entries from Global Security Intelligence Lists



- Note**
- In multi-domain deployments, the names of these lists may not be "Global." For more information, see [Security Intelligence Lists and Multitenancy, on page 33](#).
 - To add entries to these lists, see [Add Entries to Global Security Intelligence Lists, on page 34](#).

Procedure

- Step 1** Choose **Objects > Object Management**.
- Step 2** Click **Security Intelligence**.
- Step 3** Click the appropriate option:
- **Network Lists and Feeds** (for IP addresses)
 - **DNS Lists and Feeds** (for domain names)
 - **URL Lists and Feeds**
- Step 4** Click the pencil beside the Global Block or Global Do-Not-Block list.
- Step 5** Click the trash button beside the entry to delete.

List and Feed Updates for Security Intelligence

List and feed updates replace the existing list or feed file with the contents of the new file. Contents of existing and new files are not merged.

If the system downloads a corrupt feed or a feed with no recognizable entries, the system continues using the old feed data (unless it is the first download). However, if the system can recognize even one entry in the feed, it uses the entries it can recognize.

By default, each feed updates the Management Center every two hours; you can modify this frequency. Any updates the Management Center receives are passed immediately to managed devices. In addition, managed devices poll the FMC every 30 minutes for changes. You cannot modify this frequency.

In a multidomain deployment, the system-provided feeds belong to the Global domain and can be modified only by an administrator in that domain. You can modify the update frequency for custom feeds belonging to your domain.



To modify feed update intervals, see [Changing the Update Frequency for Security Intelligence Feeds, on page 36](#).

Changing the Update Frequency for Security Intelligence Feeds

You can specify the intervals at which the Firepower Management Center updates Security Intelligence Feeds.

For details about feed updates, see [List and Feed Updates for Security Intelligence, on page 36](#).

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**.
-

Custom Security Intelligence Lists and Feeds

Custom Lists and Feeds: Requirements

List and Feed Formatting

Each list or feed must be a simple text file no larger than 500MB. List files must have the .txt extension. Include one entry or comment per line: one IP address, one URL, one domain name.



Tip The number of entries you can include is limited by the maximum size of the file. For example, a URL list with no comments and an average URL length of 100 characters (including Punycode or percent Unicode representations and newlines) can contain more than 5.24 million entries.

In a DNS list entry, you can specify an asterisk (*) wildcard character for a domain label. All labels match the wildcard. For example, an entry of `www.example.*` matches both `www.example.com` and `www.example.co`.

If you add comment lines within the source file, they must start with the pound (#) character. If you upload a source file with comments, the system removes your comments during upload. Source files you download contain all your entries without your comments.

Feed Requirements

When you configure a feed, you specify its location using a URL; the URL cannot be Punycode-encoded.

If you use an MD5 checksum, the checksum must be stored in a simple text file with only the checksum. Comments are not supported.

URL Lists and Feeds: URL Syntax and Matching Criteria

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 the following, which have the matching behavior as described:

- 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`, and `www.example.com/def` -- but NOT `example.co.uk` or `examplexyz.com` or `example.com.malicious-site.com`

You can also include an entire URL path, such as

`https://www.cisco.com/c/en/us/products/security/firewalls/index.html`

- 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 40](#).

Custom Security Intelligence Feeds

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 Firepower Management Center appliances in your deployment using one source list.



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, respectively, and a default value of `any` for the **Source Networks** and **Destination Networks**.

You also 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 time the system downloaded the feed, the system does not need to re-download it. You may want to use MD5 checksums for internal feeds, especially if they are large.



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 the most up-to-date, relevant data.

Manually updating Security Intelligence feeds updates all feeds, including the Intelligence Feeds.

See complete requirements at [Custom Lists and Feeds: Requirements, on page 36](#).

Creating Security Intelligence Feeds

You must have the Threat license (for FTD devices) or the Protection license (all other device types).

Procedure

- Step 1** Choose **Objects > Object Management**.
- Step 2** Expand the **Security Intelligence** node, then choose a feed type you want to add.
- Step 3** Click the option appropriate to the feed type you chose above:
- **Add Network Lists and Feeds** (for IP addresses)
 - **Add DNS Lists and Feeds**
 - **Add URL Lists and Feeds**
- Step 4** Enter a **Name** for the feed.
- In a multidomain deployment, 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.
- Step 5** Choose **Feed** from the **Type** drop-down list.
- Step 6** Enter a **Feed URL**.
- Step 7** (Optional) Enter an **MD5 URL**.
- This is used to determine whether the feed contents have changed since the last update, so the system does not download unchanged feeds.
- Step 8** Choose an **Update Frequency**.
- Step 9** Click **Save**.
- Unless you disabled feed updates, the system attempts to download and verify the feed.
-

Manually Updating Security Intelligence Feeds

You must have the Threat license (for FTD 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**.
- Step 2** Expand the **Security Intelligence** node, then choose a feed type.
- Step 3** Click **Update Feeds**, then confirm.
- Step 4** Click **OK**.
-

After the Firepower 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

Security Intelligence lists are simple static lists of IP addresses and address blocks, URLs, or domain names that you manually upload to the system. Custom lists are useful if you want to augment and fine-tune feeds or one of the global lists, for a single Firepower Management Center's managed devices.

For example, if a reputable feed improperly blocks your access to vital resources but is overall useful to your organization, you can create a custom Do Not Block list that contains only the improperly classified IP addresses, rather than removing the IP address feed object from the access control policy's Block list.



Note You cannot add address blocks to a Block or Do Not Block list using a /0 netmask in a Security Intelligence list. 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, respectively, and a default value of `any` for the **Source Networks** and **Destination Networks**.

Regarding list entry formatting, note the following:

- Netmasks for address blocks can be integers from 0 to 32 or 0 to 128, for IPv4 and IPv6, respectively.
- Unicode in domain names must be encoded in Punycode format, and are case insensitive.
- Characters in domain names are case-insensitive.
- Unicode in URLs should be encoded in percent-encoding format.
- Characters in URL subdirectories are case-sensitive.
- List entries that start with the pound sign (#) are treated as comments.
- See additional formatting requirements at [Custom Lists and Feeds: Requirements, on page 36](#).

Regarding matching list entries, note the following:

- The system matches sub-level domains if a higher-level domain exists in a URL or DNS list. For example, if you add `example.com` to a DNS list, the system matches both `www.example.com` and `test.example.com`.
- The system does not perform DNS lookups (forward or reverse) on DNS or URL list entries. For example, if you add `http://192.168.0.2` to a URL list, and it resolves to `http://www.example.com`, the system only matches `http://192.168.0.2`, not `http://www.example.com`.

Uploading New Security Intelligence Lists to the Firepower 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**.
- Step 2** Expand the **Security Intelligence** node, then choose a list type.
- Step 3** 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 4 Enter a **Name**.

In a multidomain deployment, 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.

Step 5 From the **Type** drop-down list, choose **List**.

Step 6 Click **Browse** to browse to the list `.txt` file, then click **Upload**.

Step 7 Click **Save**.

What to do next

If an active policy references your object, deploy configuration changes, see [Deploy Configuration Changes](#)


Updating Security Intelligence Lists


In a multidomain deployment, the system displays objects created in the current domain, which you can edit. It also displays objects created in ancestor domains, which in most cases you cannot edit. To view and edit objects in a descendant domain, switch to that domain.

Procedure

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

Step 2 Expand the **Security Intelligence** node, then choose a list type.

Step 3 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 4 If you need a copy of the list to edit, click **Download**, then follow your browser's prompts to save the list as a text file.

Step 5 Make changes to the list as necessary.

Step 6 On the Security Intelligence pop-up window, click **Browse** to browse to the modified list, then click **Upload**.

Step 7 Click **Save**.

What to do next

If an active policy references your object, deploy configuration changes, see [Deploy Configuration Changes](#).

Sinkhole Objects

A sinkhole object represents either a DNS server that gives non-routeable addresses for all domain names within the sinkhole, or an IP address that does not resolve to a server. You can reference the sinkhole object within a DNS policy rule to redirect matching traffic to the sinkhole. You must assign the object both an IPv4 address and an IPv6 address.

Creating Sinkhole Objects

You must have the Threat license (for FTD devices) or the Protection license (all other device types).

Procedure

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

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

Step 3 Click **Add Sinkhole**.

Step 4 Enter a **Name**.

In a multidomain deployment, 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.

Step 5 Enter the **IPv4 Address** and **IPv6 Address** of your sinkhole.

Step 6 You have the following options:

- If you want to redirect traffic to a sinkhole server, choose **Log Connections to Sinkhole**.
- If you want to redirect traffic to a non-resolving IP address, choose **Block and Log Connections to Sinkhole**.

Step 7 If you want to assign an Indication of Compromise (IoC) type to your sinkhole, choose one from the **Type** drop-down.

Step 8 Click **Save**.

File Lists

If you use AMP for Networks, and the AMP cloud incorrectly identifies a file's disposition, you can add the file to a *file list* to better detect the file in the future. These files are specified using SHA-256 hash values. Each file list can contain up to 10000 unique SHA-256 values.

There are two predefined categories of file lists:

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.

In a multidomain deployment, a clean list and custom detection list is present for each domain. In lower-level domains, you can view but not modify ancestor's lists.

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 a **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 and the custom detection list.

Source Files for File Lists

You can add multiple SHA-256 values to a file list by uploading a comma-separated value (CSV) source file containing a list of SHA-256 values and descriptions. The Firepower Management Center validates the contents and populates the file list with valid SHA-256 values.

The source file must be a simple text file with a .csv file name extension. Any header must start with a pound sign (#); it is treated as a comment and not uploaded. Each entry should contain a single SHA-256 value followed by a description and end with either the LF or CR+LF Newline character. The system ignores any additional information in the entry.

Note the following:

- Deleting a source file from the file list also removes all associated SHA-256 hashes from the file list.
- You cannot upload multiple files to a file list if the successful source file upload results in the file list containing more than 10000 distinct SHA-256 values.
- The system truncates descriptions exceeding 256 characters to the first 256 characters on upload. If the description contains commas, you must use an escape character (\,). If no description is included, the source file name is used instead.
- All non-duplicate SHA-256 values are added to the file list. If a file list contains a SHA-256 value, and you upload a source file containing that value, the newly uploaded value does not modify the existing SHA-256 value. When viewing captured files, file events, or malware events related to the SHA-256 value, any threat name or description is derived from the individual SHA-256 value.
- The system does not upload invalid SHA-256 values in a source file.
- If multiple uploaded source files contain an entry 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 one.
- You cannot directly edit a source file within the object manager. To make changes, you must first modify your source file directly, delete the copy on the system, then upload the modified source file.
- The number of entries associated with a source file refers to the number of distinct SHA-256 values. If you delete a source file from a file list, the total number of SHA-256 entries the file list contains decreases by the number of valid entries in the source file.

Adding Individual SHA-256 Values to File Lists

You must have the Malware license for this procedure.

You can submit a file's SHA-256 value to add it to a file list. You cannot add duplicate SHA-256 values.

In a multidomain deployment, the system displays objects created in the current domain, which you can edit. It also displays objects created in ancestor domains, which in most cases you cannot edit. To view and edit objects in a descendant domain, switch to that domain.


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.

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.



Uploading Individual Files to File Lists

You must have the Malware 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 Firepower Management Center for analysis; the system calculates the file's SHA-256 value and adds the file to the list. The system does not enforce a limit on the size of files for SHA-256 calculation.

In a multidomain deployment, the system displays objects created in the current domain, which you can edit. It also displays objects created in ancestor domains, which in most cases you cannot edit. To view and edit objects in a descendant domain, switch to that domain.

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** Optionally, enter a description of the file in the **Description** field. If you do not enter a description, the file name is used for the description on upload.
 - Step 6** Click **Browse**, and choose a file to upload.
 - Step 7** Click **Calculate and Add SHA**.
 - Step 8** Click **Save**.
-

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.


Uploading Source Files to File Lists

You must have the Malware license for this procedure.

In a multidomain deployment, the system displays objects created in the current domain, which you can edit. It also displays objects created in ancestor domains, which in most cases you cannot edit. To view and edit objects in a descendant domain, switch to that domain.

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** Optionally, 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**.

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 no longer queries the AMP cloud for files on the list.





Editing SHA-256 Values in File Lists

You must have the Malware license for this procedure.

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

In a multidomain deployment, the system displays objects created in the current domain, which you can edit. It also displays objects created in ancestor domains, which in most cases you cannot edit. To view and edit objects in a descendant domain, switch to that domain.

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.

Downloading Source Files from File Lists


You must have the Malware license for this procedure.


In a multidomain deployment, the system displays objects created in the current domain, which you can edit. It also displays objects created in ancestor domains, which in most cases you cannot edit. To view and edit objects in a descendant domain, switch to that domain.


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**.

Cipher Suite Lists

A cipher suite list is an object comprised of several cipher suites. Each predefined cipher suite value represents a cipher suite used to negotiate an SSL- or TLS-encrypted session. You can use cipher suites and cipher suite lists in SSL rules to control encrypted traffic based on whether the client and server negotiated the SSL session using that cipher suite. If you add a cipher suite list to an SSL rule, SSL sessions negotiated with any of the cipher suites in the list match the rule.




Note Although you can use cipher suites in the web interface in the same places as cipher suite lists, you cannot add, modify, or delete cipher suites.

Creating Cipher Suite Lists

You can use these objects with any device type except NGIPSv.

Procedure

-
- Step 1** Choose **Objects > Object Management**.
 - Step 2** Choose **Cipher Suite List** from the list of object types.
 - Step 3** Click **Add Cipher Suites**.
 - Step 4** Enter a **Name**.
- In a multidomain deployment, 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.
- Step 5** Choose one or more cipher suites from the **Available Ciphers** list.
 - Step 6** Click **Add**.
 - Step 7** Optionally, click **Delete** () next to any cipher suites in the **Selected Ciphers** list that you want to remove.
 - Step 8** Click **Save**.
-

What to do next

- If an active policy references your object, deploy configuration changes; see [Deploy Configuration Changes](#).

Distinguished Name Objects

Each distinguished name object represents the [distinguished name](#) for a public key certificate's subject or issuer. You can use distinguished name objects and groups 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 group* is a named collection of existing distinguished name objects.)

The distinguished name can consist of country code, common name, organization, and organizational unit, but typically consists of a common name only. For example, the common name in the certificate for `https://www.cisco.com` is `cisco.com`. The certificate can contain multiple Subject Alternative Names (SANs) you can use as DNs in a rule condition. For detailed information about SANs, see [RFC 528, 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 Firepower 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 the following table, separated by commas.

Table 3: 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

Important notes about DN rule conditions

- The first time the system detects an encrypted session to a new server, DN data is not available for ClientHello processing, which *might* result in an undecrypted first session.
- 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 the following table for examples.

Table 4: 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

Attribute	Matches	Does Not Match
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 we recommend wildcards in these cases.

Creating Distinguished Name Objects

You can use these objects with any device type except NGIPSv.

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**.

In a multidomain deployment, 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.

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 Name Objects, on page 48](#) 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](#).

PKI Objects

PKI Objects for SSL Application

PKI objects represent the public key certificates and paired private keys required to support your deployment. Internal and trusted CA objects consist of certificate authority (CA) certificates; internal CA objects also contain the private key paired with the certificate. Internal and external certificate objects consist of server certificates; internal certificate objects also contain the private key paired with the certificate.

If you use trusted certificate authority objects and internal certificate objects to configure a connection to ISE, you can use ISE as an identity source.

If you use internal certificate objects to configure captive portal, the system can authenticate the identity of your captive portal device when connecting to users' web browsers.

If you use trusted certificate authority objects to configure realms, you can configure 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 Firepower 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.

Internal Certificate Authority Objects

Each internal certificate authority (CA) object you configure represents the CA public key certificate of a CA your organization controls. The object consists of the object name, CA certificate, and paired private key. You can use internal CA objects and groups in SSL rules to decrypt outgoing encrypted traffic by re-signing the server certificate with the internal CA.



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.

You can create an internal CA object in the following 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.

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 Certificate and Private Key Import

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 the following 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.

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.



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.

Importing a CA Certificate and Private Key

You can use these objects with any device type except NGIPSv.

In a multidomain deployment, the system displays objects created in the current domain, which you can edit. It also displays objects created in ancestor domains, which in most cases you cannot edit. To view and edit objects in a descendant domain, switch to that domain.

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**.
- In a multidomain deployment, 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.
- 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](#).

Generating a New CA Certificate and Private Key

You can use these objects with any device type except NGIPSV.

You can configure an internal CA object by providing identification information to generate a self-signed RSA-based CA certificate and private key.

The generated CA certificate is valid for ten years. The Valid From date is a week before generation.

In a multidomain deployment, the system displays objects created in the current domain, which you can edit. It also displays objects created in ancestor domains, which in most cases you cannot edit. To view and edit objects in a descendant domain, switch to that domain.

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**.
- In a multidomain deployment, 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.
- Step 5** Enter the identification attributes.

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

New Signed Certificates

You can configure an internal CA object by obtaining a signed certificate from a CA. This involves two steps:

- Provide identification information to configure the internal CA object. This generates an unsigned certificate and paired private key, and creates a certificate signing request (CSR) to a CA you specify.
- After the CA issues the signed certificate, upload it to the internal CA object, replacing the unsigned certificate.

You can only reference an internal CA object in an SSL rule if it contains a signed certificate.

Creating an Unsigned CA Certificate and CSR

You can use these objects with any device type except NGIPSv.

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**.

In a multidomain deployment, 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.

- 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

- You must upload a signed certificate issued by a CA as described in [Uploading a Signed Certificate Issued in Response to a CSR](#), on page 54


Uploading a Signed Certificate Issued in Response to a CSR

You can use these objects with any device type except NGIPSv.

In a multidomain deployment, the system displays objects created in the current domain, which you can edit. It also displays objects created in ancestor domains, which in most cases you cannot edit. To view and edit objects in a descendant domain, switch to that domain.

Once uploaded, the signed certificate can be referenced in SSL rules.

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 Certificate and Private Key Downloads

You can back up or transfer a CA certificate and paired private key by downloading a file containing the certificate and key information from an internal CA object.



Caution Always store downloaded key information in a secure location.

The system encrypts the private key stored in an internal CA object with a randomly generated key before saving it to disk. If you download a certificate and private key from an internal CA object, the system first decrypts the information before creating a file containing the certificate and private key information. You must then provide a password the system uses to encrypt the downloaded file.



Caution Private keys downloaded as part of a system backup are decrypted, then stored in the unencrypted backup file.



Downloading a CA Certificate and Private Key

You can use these objects with any device type except NGIPsv.

In a multidomain deployment, the system displays objects created in the current domain, which you can edit. It also displays objects created in ancestor domains, which in most cases you cannot edit. To view and edit objects in a descendant domain, switch to that domain.

You can download CA certificates for both the current domain and ancestor domains.

Procedure

- Step 1** Choose **Objects > Object Management**.
- Step 2** Expand the **PKI** node, and choose **Internal CAs**.
- Step 3** Next to the internal CA object whose certificate and private key you want to download, click **Edit** ().
- In a multidomain deployment, click **View** () to download the certificate and private key for an object in an ancestor domain.
- 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

Each trusted certificate authority (CA) object you configure represents a CA public key certificate belonging to a trusted CA. The object consists of the object name and CA public key certificate. You can use external CA objects and groups in:

- your SSL policy to control traffic encrypted with a certificate signed either by the trusted CA, or any CA within the chain of trust.
- your realm configurations to establish secure connections to LDAP or AD servers.
- your ISE 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 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 Object

You can configure an external CA object by uploading an X.509 v3 CA certificate. You can upload a file encoded in one of the following 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 information.

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

Adding a Trusted CA Object

You can use these objects with any device type except NGIPSv.

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**.

In a multidomain deployment, 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.
 - 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**.
-

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

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. You can upload files encoded in one of the following supported formats:

- 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 integration configuration.

Adding a Certificate Revocation List to a Trusted CA Object

You can use these objects with any device type except NGIPSv.

In a multidomain deployment, the system displays objects created in the current domain, which you can edit. It also displays objects created in ancestor domains, which in most cases you cannot edit. To view and edit objects in a descendant domain, switch to that domain.




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

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** to upload a DER or PEM-encoded CRL file.

Step 5 Click **OK**.

What to do next

- If an active policy references your object, deploy configuration changes; see [Deploy Configuration Changes](#).

External Certificate Objects

Each external certificate object you configure represents a server public key certificate that does not belong to your organization. The object consists of the object name and certificate. You can use external certificate objects and groups in SSL rules to control traffic encrypted with the server certificate. For example, you can upload a self-signed server certificate that you trust, but cannot verify with a trusted CA certificate.

You can configure an external certificate object by uploading an X.509 v3 server certificate. You can upload a file in one of the following supported formats:

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

You can upload only files that contains proper server certificate information; the system validates the file before saving the object. If the certificate is encoded in the PEM format, you can also copy and paste the information.

Adding External Certificate Objects

You can use these objects with any device type except NGIPSv.

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**.
- In a multidomain deployment, 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.
- 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

Each internal certificate object you configure represents a server public key certificate belonging to your organization. The object consists of the object name, public key certificate, and paired private key. 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 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 the following 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.

Adding Internal Certificate Objects

You can use these objects with any device type except NGIPSv.

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**.

In a multidomain deployment, 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.
 - 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**.
-