Configuring Special Actions for Application Inspections (Inspection Policy Map)

Modular Policy Framework lets you configure special actions for many application inspections. When you enable an inspection engine in the service policy, you can also optionally enable actions as defined in an *inspection policy map*. When the inspection policy map matches traffic within the service policy for which you have defined an inspection action, then that subset of traffic will be acted upon as specified (for example, dropped or rate-limited).

This chapter includes the following sections:

Information About Inspection Policy Maps

See the "Configuring Application Layer Protocol Inspection" section on page 10-7 for a list of applications that support inspection policy maps.

An inspection policy map consists of one or more of the following elements. The exact options available for an inspection policy map depends on the application.

- Traffic matching option—You can define a traffic matching option directly in the inspection policy map to match application traffic to criteria specific to the application, such as a URL string, for which you then enable actions.
  - Some traffic matching options can specify regular expressions to match text inside a packet. Be sure to create and test the regular expressions before you configure the policy map, either singly or grouped together in a regular expression class map.
- Inspection class map—An inspection class map includes multiple traffic matching options. You then identify the class map in the policy map and enable actions for the class map as a whole. The difference between creating a class map and defining the traffic match directly in the inspection

policy map is that you can create more complex match criteria and you can reuse class maps. However, you cannot set different actions for different matches. **Note:** Not all inspections support inspection class maps.

- Parameters—Parameters affect the behavior of the inspection engine.

Guidelines and Limitations

- HTTP inspection policy maps—If you modify an in-use HTTP inspection policy map, you must remove and reapply the inspection policy map action for the changes to take effect. For example, if you modify the “http-map” inspection policy map, you must remove, apply changes, and readd the inspection policy map to the service policy.
- All inspection policy maps—If you want to exchange an in-use inspection policy map for a different map name, you must remove, apply changes, and readd the new inspection policy map to the service policy.
- You can specify multiple inspection class maps or direct matches in the inspection policy map.

  If a packet matches multiple different matches, then the order in which the ASA applies the actions is determined by internal ASA rules, and not by the order they are added to the inspection policy map. The internal rules are determined by the application type and the logical progression of parsing a packet, and are not user-configurable. For example for HTTP traffic, parsing a Request Method field precedes parsing the Header Host Length field; an action for the Request Method field occurs before the action for the Header Host Length field.

  If an action drops a packet, then no further actions are performed in the inspection policy map. For example, if the first action is to reset the connection, then it will never match any further match criteria. If the first action is to log the packet, then a second action, such as resetting the connection, can occur.

  If a packet matches multiple match criteria that are the same, then they are matched in the order they appear in the policy map.

  A class map is determined to be the same type as another class map or direct match based on the lowest priority match option in the class map (the priority is based on the internal rules). If a class map has the same type of lowest priority match option as another class map, then the class maps are matched according to the order they are added to the policy map. If the lowest priority match for each class map is different, then the class map with the higher priority match option is matched first.

Default Inspection Policy Maps

DNS inspection is enabled by default, using the preset_dns_map inspection class map:

- The maximum DNS message length is 512 bytes.
- The maximum client DNS message length is automatically set to match the Resource Record.
- DNS Guard is enabled, so the ASA tears down the DNS session associated with a DNS query as soon as the DNS reply is forwarded by the ASA. The ASA also monitors the message exchange to ensure that the ID of the DNS reply matches the ID of the DNS query.
- Translation of the DNS record based on the NAT configuration is enabled.
- Protocol enforcement is enabled, which enables DNS message format check, including domain name length of no more than 255 characters, label length of 63 characters, compression, and looped pointer check.

Note There are other default inspection policy maps such as **_default_esmtp_map**. For example, an ESMTP inspection rule implicitly uses the policy map "_default_esmtp_map."

Defining Actions in an Inspection Policy Map

When you enable an inspection engine in the service policy, you can also optionally enable actions as defined in an inspection policy map.

Detailed Steps

Step 1 (Optional) Create an inspection class map. Alternatively, you can identify the traffic directly within the policy map. See the "Identifying Traffic in an Inspection Class Map" section on page 2-3.

Step 2 (Optional) For policy map types that support regular expressions, create a regular expression. See the "Configuring Regular Expressions" section on page 20-20 in the general operations configuration guide.

Step 3 Choose **Configuration > Firewall > Objects > Inspect Maps** .

Step 4 Choose the inspection type you want to configure.

Step 5 Click **Add** to add a new inspection policy map.

Step 6 Follow the instructions for your inspection type in the inspection chapter.

Identifying Traffic in an Inspection Class Map

This type of class map allows you to match criteria that is specific to an application. For example, for DNS traffic, you can match the domain name in a DNS query.

A class map groups multiple traffic matches (in a match-all class map), or lets you match any of a list of matches (in a match-any class map). The difference between creating a class map and defining the traffic match directly in the inspection policy map is that the class map lets you group multiple match commands, and you can reuse class maps. For the traffic that you identify in this class map, you can specify actions such as dropping, resetting, and/or logging the connection in the inspection policy map. If you want to perform different actions on different types of traffic, you should identify the traffic directly in the policy map.

Restrictions

Not all applications support inspection class maps.

Detailed Steps

Step 1 Choose **Configuration > Firewall > Objects > Class Maps** .

Step 2 Choose the inspection type you want to configure.

Step 3 Click **Add** to add a new inspection class map.

Step 4 Follow the instructions for your inspection type in the inspection chapter.

Where to Go Next

To use an inspection policy, see Chapter 1, "Configuring a Service Policy."

Feature History for Inspection Policy Maps

Table 2-1 lists the release history for this feature.

Table 2-1 Feature History for Service Policies

Feature Name	Releases	Feature Information
Inspection policy maps	7.2(1)	The inspection policy map was introduced. The following command was introduced: **class-map type inspect**.
Regular expressions and policy maps	7.2(1)	Regular expressions and policy maps were introduced to be used under inspection policy maps. The following commands were introduced: **class-map type regex**, **regex**, **match regex**.
Match any for inspection policy maps	8.0(2)	The **match any** keyword was introduced for use with inspection policy maps: traffic can match one or more criteria to match the class map. Formerly, only **match all** was available.