Policy Classification Engine

The Policy Classification Engine feature helps configure device-based policies and client (network endpoint) profiling and enforces a per user or per device policy on a network. The policy classification engine enables bring-your-own-device (BYOD) deployments integrate user or wireless device policies into the wireless controller. This module explains how to configure policies and apply them to a wireless LAN (WLAN).

- Restrictions for Policy Classification Engine, on page 1
- Information About Policy Classification Engine, on page 1
- How to Configure Policy Classification Engine, on page 2
- Configuration Examples for Policy Classification Engine, on page 5
- Feature History for Policy Classification Engine, on page 5

Restrictions for Policy Classification Engine

Interface templates are not valid on wireless sessions.

Information About Policy Classification Engine

Policy Classification Engine Overview

The Policy Classification Engine feature helps configure device-based policies and client (network endpoint) profiling and enforces a per user or per device policy on a network.

You can configure sets of different policies that can be used for lookup and sequential matching. A policy is matched based on the configured policy statement. Use policies to profile devices based on the Dynamic Host Control Protocol (DHCP) or HTTP to identify end devices in a network. You can enforce specific policies at network endpoints.

The device (switch; for example, Cisco Catalyst 3850 Wireless LAN Controller) uses these attributes and predefined classification profiles to identify devices.

Policies are configured based on the following parameters:

- Device—Types of end devices. Examples are Windows machines, smart phones, Apple device like iPads, iPhones, and so on.
- Regular expressions

- User role—The user type or user group to which an user belongs. Examples are students, employees, and so on.
- Username—Login credentials entered by users.
- Time-of-day—The time-of-day when endpoints are allowed into a network.
- OUI—The MAC address that identifies the Organizational Unique Identifier (OUI).
- MAC address—The MAC address of the endpoint.

Once the device (switch) has a match corresponding to the policy parameters per end point, a policy is added. Policy enforcement is based on the following session attributes:

- VLAN—User-defined VLAN
- Access control list (ACL)
- Session timeout value—User-defined timeout for client sessions
- Quality of service (QoS)

You can configure policies and based on the session attributes, enforce these policies on end points.

How to Configure Policy Classification Engine

Configuring Policies in Cisco Identity Based Networking Services

To configure policies, perform the following tasks:

1. Configure a service template.
2. Configure an interface template.
3. Create a parameter map.
4. Create a policy map.
5. Apply the policy on a wireless LAN (WLAN).

Configuring a Subscriber Parameter Map

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**parameter-map type subscriber attribute-to-service** *parameter-map-name* **Example:** `Device(config)# parameter-map type subscriber attribute-to-service param-map`	Configures a subscriber parameter map and enters parameter-map filter configuration mode.
Step 4	*priority-number* **map device-type eq** *device-type* **oui eq** *MAC-address* **Example:** `Device(config-parameter-map-filter)# 1 map device-type eq "Cisco-IP-Phone-9971" oui eq "08.cc.68"`	Maps the priority and the Organizationally Unique Identifier (OUI) of the configured device, and enters parameter-map filter submode configuration mode.
Step 5	*action-number* **interface-template** *interface-template-name* **Example:** `Device(config-parameter-map-filter-submode)# 2 interface-template IP-PHONE-INTERFACE-TEMPLATE`	Maps the action number to an interface template.
Step 6	**end** **Example:** `Device(config-parameter-map-filter-submode)# end`	Exits parameter-map filter submode configuration mode and returns to privileged EXEC mode.
Step 7	**show parameter-map type subscriber attribute-to-service** *parameter-map-name* **Example:** `Device# show parameter-map type subscriber attribute-to-service parameter-map-name`	Displays information about the specified parameter map.

Example

The following is sample output from the **show parameter-map type subscriber attribute-to-service** command:

```
Device# show parameter-map type subscriber attribute-to-service param-map

Parameter-map name: param-map
 Map: 1 map device-type eq "Cisco-IP-Phone-9971" oui eq "08.cc.68"
  Action(s):
   2 interface-template IP-PHONE-INTERFACE-TEMPLATE
```

Configuring a Subscriber Policy Map

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**policy-map type control subscriber** *policy-map-name* **Example:** `Device(config)# policy-map type control subscriber pmap`	Defines a control policy for subscriber sessions and enters control policy-map event configuration mode.
Step 4	**event identity-update** {**match-all** | **match-first**} **Example:** `Device(config-event-control-policymap)# event identity-update match-all`	Specifies the event type that triggers actions in a control policy if conditions are met, and enters control policy-map class configuration mode.
Step 5	*priority-number* **class always** {**do-all** | **do-until-failure** | **do-until-success**} **Example:** `Device(config-class-control-policymap)# 1 class always do-until-failure`	Associates a control class with one or more actions in a control policy and enters control policy-map action configuration mode.
Step 6	*action-number* **map attribute-to-service table** *parameter-map-name* **Example:** `Device(config-action-control-policymap)# 2 map attribute-to-service table param-map`	Maps identity-update attribute to an autoconf template.
Step 7	**end** **Example:** `Device(config-action-control-policymap)# end`	Exits control policy-map action configuration mode and returns to privileged EXEC mode.
Step 8	**show policy-map type control subscriber** *policy-map-name* **Example:**	Displays information and statistics about the control policies.

	Command or Action	Purpose
	`Device# show policy-map type control subscriber pmap`	

Example

The following is sample output from the **show policy-map type control subscriber** command:

```
Device# show policy-map type control subscriber pmap

show policy-map type control subscriber pmap
policy-map
  event identity-update match-all
    1 class always do-until-failure
      1 map attribute-to-service table param-map
```

Configuration Examples for Policy Classification Engine

Example: Configuring a Subscriber Parameter Map

```
Device# configure terminal
Device(config)# parameter-map type subscriber attribute-to-service param-map
Device(config-parameter-map-filter)# 1 map device-type eq "Cisco-IP-Phone-9971" oui "eq
08.cc.68"
Device(config-parameter-map-filter-submode)# 2 interface-template IP-PHONE-INTERFACE-TEMPLATE
Device(config-parameter-map-filter-submode)# end
```

Example: Configuring a Subscriber Policy Map

```
Device# configure terminal
Device(config)# policy-map type control subscriber pmap
Device(config-event-control-policymap)# event identity-update match-all
Device(config-class-control-policymap)# 1 class always do-until-failure
Device(config-action-control-policymap)# 2 map attribute-to-service table param-map
Device(config-action-control-policymap)# end
```

Feature History for Policy Classification Engine

This table provides release and related information for the features explained in this module.

These features are available in all the releases subsequent to the one they were introduced in, unless noted otherwise.

Table 1: Feature History for Policy Classification Engine

Release	Feature Name	Feature Information
Cisco IOS XE Everest 16.5.1a	Policy Classification Engine	The Policy Classification Engine feature helps configure device-based policies and client (network endpoint) profiling and enforces a per user or per device policy on a network. The policy classification engine enables bring-your-own-device (BYOD) deployments integrate user or wireless device policies into the wireless controller. Support for this feature was introduced only on the C9500-12Q, C9500-16X, C9500-24Q, C9500-40X models of the Cisco Catalyst 9500 Series Switches
Cisco IOS XE Fuji 16.8.1a	Policy Classification Engine	Support for this feature was introduced only on the C9500-32C, C9500-32QC, C9500-48Y4C, and C9500-24Y4C models of the Cisco Catalyst 9500 Series Switches.
Cisco IOS XE Cupertino 17.7.1	Policy Classification Engine	This feature was implemented on the C9500X-28C8D model.
Cisco IOS XE Dublin 17.10.1b	Policy Classification Engine	This feature was implemented on the C9500X-60L4D model.

Use the Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to https://cfnng.cisco.com.