CONTENTS

CHAPTER 1

Identity-Based Networking Services Overview 1
Finding Feature Information 1
Information About Identity-Based Networking Services 1
Understanding Identity-Based Networking Services 1
Features in Identity-Based Networking Services 2
Benefits of Identity-Based Networking Services 2
IP Device Tracking 3
Additional References 3
Feature Information for Identity-Based Networking Services Overview 4

CHAPTER 2

Change of Authorization Support 5
Finding Feature Information 5
Information About CoA Support 5
RADIUS Change-of-Authorization Support 5
Session Identification 6
CoA Activate Service Command 7
CoA Deactivate Service Command 7
CoA Bounce Host Port Command 8
CoA Disable Host Port Command 8
CoA Session Query Command 8
CoA Session Reauthenticate Command 9
CoA Session Terminate Command 9
Additional References 10
Feature Information for CoA Support 11

CHAPTER 3

Web Authentication Redirection to Original URL 13
Information About Web Authentication Redirection to Original URL 13
Web Authentication Redirection to Original URL Overview 13
CHAPTER 4

Configuring Identity Control Policies 17

Finding Feature Information 17
Information About Identity Control Policies 17
Identity-Based Networking Services Configuration 17
Concurrent Authentication Methods 18
Configuration Display Mode 18
Control Policies for Identity-Based Networking Services 19
Control Policy Configuration Overview 20
Parameter Maps for Identity-Based Networking Services 20
Per User Inactivity Handling Across Methods 20
How to Configure Identity Control Policies 21
Enabling the Display Mode for Identity-Based Networking Services 21
Configuring a Control Class 22
Configuring a Control Policy 25
Applying a Control Policy to an Interface 30
Configuring Authentication Features on Ports 31
Configuring a Parameter Map for Web-Based Authentication 32
Configuration Examples for Identity Control Policies 36
Example: Configuring Control Policy for Concurrent Authentication Methods 36
Example: Configuring Control Policy for Sequential Authentication Methods 37
Example: Configuring Parameter Maps 38
Additional References 39
Feature Information for Identity Control Policies 40

CHAPTER 5

Policy Classification Engine 43

Finding Feature Information 43
Restrictions for Policy Classification Engine 43
Information About Policy Classification Engine 44
Policy Classification Engine Overview 44
How to Configure Policy Classification Engine 44
Configuring Policies in Identity-Based Networking Services 44
Configuring a Subscriber Parameter Map 45
Identity-Based Networking Services Overview

Identity-Based Networking Services provides a policy and identity-based framework in which edge devices can deliver flexible and scalable services to subscribers. This module provides information about what Identity-Based Networking Services is and its features and benefits.

- Finding Feature Information, page 1
- Information About Identity-Based Networking Services, page 1
- Additional References, page 3
- Feature Information for Identity-Based Networking Services Overview, page 4

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About Identity-Based Networking Services

Understanding Identity-Based Networking Services

Identity-Based Networking Services provides an identity-based approach to access management and subscriber management. It offers a consistent way to configure features across technologies, a command interface that allows easy deployment and customization of features, and a robust policy control engine with the ability to apply policies defined locally or received from an external server to enforce policy in the network.
The figure below illustrates a typical deployment of Identity-Based Networking Services in a physically distributed enterprise with a campus, branch offices, and remote workers.

**Figure 1: Sample Deployment**

---

**Features in Identity-Based Networking Services**

Identity-Based Networking Services includes the following features:

- Cisco common classification policy language (C3PL)-based identity configuration
- Concurrent authentication methods on a single session, including IEEE 802.1x (dot1x), MAC authentication bypass (MAB), and web authentication
- Downloadable identity service templates
- Extended RADIUS change of authorization (CoA) support for querying, reauthenticating, and terminating a session, port shutdown and port bounce, and activating and deactivating an identity service template.
- Local authentication using Lightweight Directory Access Protocol (LDAP)
- Locally defined identity control policies
- Locally defined identity service templates
- Per-user inactivity handling across methods

**Benefits of Identity-Based Networking Services**

Identity-based solutions are essential for delivering access control for disparate groups such as employees, contractors, and partners while maintaining low operating expenses. Identity-Based Networking Services provides a consistent approach to operational management through a policy and identity-based infrastructure leading to faster deployment of new features and easier management of switches.

Identity-Based Networking Services provides the following benefits:

- An identity-based framework for session management.
- A robust policy control engine to apply policies defined locally or received from an external AAA server.
• Faster deployment and customization of features across access technologies.
• A simpler and consistent way to configure features across access methods, platforms, and application domains.

**IP Device Tracking**

IP device tracking can be configured using the Switch Integrated Security Features (SISF) policy. Use the tracking enable command in device tracking configuration mode, to configure device tracking using SISF policy. Use the `show device-tracking` command to display the device tracking configuration.

The following is the sample configuration for device tracking.

```
Device(config)# device-tracking policy sisf_policy
Device(config-device-tracking)# tracking enable
Device(config-device-tracking)# exit
Device(config)# interface GigabitEthernet 3/0/1
Device(config-if)# switchport mode access
Device(config-if)# device-tracking attach-policy sisf_policy
Device(config-if)# end
```

**Additional References**

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS commands</td>
<td>Cisco IOS Master Command List, All Releases</td>
</tr>
<tr>
<td>Identity-Based Networking Services commands</td>
<td>Cisco IOS Identity-Based Networking Services Command Reference</td>
</tr>
<tr>
<td>Address Resolution Protocol (ARP) commands</td>
<td>Cisco IOS IP Addressing Services Command Reference</td>
</tr>
<tr>
<td>ARP configuration tasks</td>
<td>IP Addressing - ARP Configuration Guide</td>
</tr>
<tr>
<td>Authentication, authorization, and accounting (AAA) configuration tasks</td>
<td>Authentication Authorization and Accounting Configuration Guide</td>
</tr>
<tr>
<td>AAA commands</td>
<td>Cisco IOS Security Command Reference</td>
</tr>
</tbody>
</table>

**Standards and RFCs**

<table>
<thead>
<tr>
<th>Standard/RFC</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFC 5176</td>
<td>Dynamic Authorization Extensions to RADIUS</td>
</tr>
</tbody>
</table>
Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support and Documentation website provides online resources to</td>
<td><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></td>
</tr>
<tr>
<td>download documentation, software, and tools. Use these resources to</td>
<td></td>
</tr>
<tr>
<td>install and configure the software and to troubleshoot and resolve</td>
<td></td>
</tr>
<tr>
<td>technical issues with Cisco products and technologies. Access to most</td>
<td></td>
</tr>
<tr>
<td>tools on the Cisco Support and Documentation website requires a Cisco.com</td>
<td></td>
</tr>
<tr>
<td>user ID and password.</td>
<td></td>
</tr>
</tbody>
</table>

Feature Information for Identity-Based Networking Services Overview

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 1: Feature Information for Identity-Based Networking Services Overview

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web Authentication Support of Common Session ID</td>
<td>Cisco IOS Release 15.2(1)SY</td>
<td>Allows a single session identifier to be used for all web authentication sessions in addition to 802.1X and MAB authenticated sessions. In Cisco IOS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Release 15.2(1)SY, this feature is supported on Cisco Catalyst 6500 Series Switches</td>
</tr>
</tbody>
</table>
Change of Authorization Support

Identity-Based Networking Services supports RADIUS change of authorization (CoA) commands for session query, reauthentication, and termination, port bounce and port shutdown, and service template activation and deactivation. This module provides information about the supported CoA commands for Identity-Based Networking Services.

- Finding Feature Information, page 5
- Information About CoA Support, page 5
- Additional References, page 10
- Feature Information for CoA Support, page 11

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About CoA Support

RADIUS Change-of-Authorization Support

Cisco IOS software supports the RADIUS CoA extensions defined in RFC 5176 that are typically used in a push model to allow the dynamic reconfiguring of sessions from external AAA or policy servers. Per-session CoA requests are supported for session identification, session termination, host reauthentication, port shutdown, and port bounce. This model comprises one request (CoA-Request) and two possible response codes:

- CoA acknowledgement (ACK) [CoA-ACK]
- CoA nonacknowledgement (NAK) [CoA-NAK]
The request is initiated from a CoA client (typically a AAA or policy server) and directed to the device that acts as a listener.

The table below shows the RADIUS CoA commands and vendor-specific attributes (VSAs) supported by Identity-Based Networking Services. All CoA commands must include the session identifier between the device and the CoA client.

Table 2: RADIUS CoA Commands Supported by Identity-Based Networking Services

<table>
<thead>
<tr>
<th>CoA Command</th>
<th>Cisco VSA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Activate service</td>
<td>Cisco:Avpair=&quot;subscriber:command=activate-service&quot;</td>
</tr>
<tr>
<td></td>
<td>Cisco:Avpair=&quot;subscriber:service-name=&lt;service-name&gt;&quot;</td>
</tr>
<tr>
<td></td>
<td>Cisco:Avpair=&quot;subscriber:precedence=&lt;precedence-number&gt;&quot;</td>
</tr>
<tr>
<td></td>
<td>Cisco:Avpair=&quot;subscriber:activation-mode=replace-all&quot;</td>
</tr>
<tr>
<td>Deactivate service</td>
<td>Cisco:Avpair=&quot;subscriber:command=deactivate-service&quot;</td>
</tr>
<tr>
<td></td>
<td>Cisco:Avpair=&quot;subscriber:service-name=&lt;service-name&gt;&quot;</td>
</tr>
<tr>
<td>Bounce host port</td>
<td>Cisco:Avpair=&quot;subscriber:command=bounce-host-port&quot;</td>
</tr>
<tr>
<td>Disable host port</td>
<td>Cisco:Avpair=&quot;subscriber:command=disable-host-port&quot;</td>
</tr>
<tr>
<td>Session query</td>
<td>Cisco:Avpair=&quot;subscriber:command=session-query&quot;</td>
</tr>
<tr>
<td>Session reauthenticate</td>
<td>Cisco:Avpair=&quot;subscriber:command=reauthenticate&quot;</td>
</tr>
<tr>
<td></td>
<td>Cisco:Avpair=&quot;subscriber:reauthenticate-type=last&quot; or</td>
</tr>
<tr>
<td></td>
<td>Cisco:Avpair=&quot;subscriber:reauthenticate-type=rerun&quot;</td>
</tr>
<tr>
<td>Session terminate</td>
<td>This is a standard disconnect request and does not require a VSA.</td>
</tr>
<tr>
<td>Interface template</td>
<td>Cisco:AVpair=&quot;interface-template-name=&lt;interfacetemplate&gt;&quot;</td>
</tr>
</tbody>
</table>

**Session Identification**

For disconnect and CoA requests targeted at a particular session, the device locates the session based on one or more of the following attributes:

- Acct-Session-Id (IETF attribute #44)
- Audit-Session-Id (Cisco VSA)
- Calling-Station-Id (IETF attribute #31, which contains the host MAC address)
- IPv6 Attributes, which can be one of the following:
  - Framed-Ipv6-Prefix (IETF attribute #97) and Framed-Interface-Id (IETF attribute #96), which together create a full IPv6 address per RFC 3162
CoA Activate Service Command

The CoA activate service command can be used to activate a service template on a session. The AAA server sends the request in a standard CoA-Request message using the following VSAs:

- Cisco:Avpair="subscriber:command=activate-service"
- Cisco:Avpair="subscriber:service-name=<service-name>"
- Cisco:Avpair="subscriber:precedence=<precedence-number>"
- Cisco:Avpair="subscriber:activation-mode=replace-all"

Because this command is session-oriented, it must be accompanied by one or more of the session identification attributes described in the "Session Identification, on page 6" section. If the device cannot locate a session, it returns a CoA-NAK message with the "Session Context Not Found" error-code attribute. If the device locates a session, it initiates an activate template operation for the hosting port and a CoA-ACK is returned. If activating the template fails, a CoA-NAK message is returned with the Error-Code attribute set to the appropriate message.

If the device fails before returning a CoA-ACK to the client, the process is repeated on the new active device when the request is re-sent from the client. If the device fails after returning a CoA-ACK message to the client but before the operation is complete, the operation is restarted on the new active device.

CoA Deactivate Service Command

The CoA deactivate service command can be used to deactivate a service template on a session. The AAA server sends the request in a standard CoA-Request message using the following VSAs:

- Cisco:Avpair="subscriber:command=deactivate-service"
- Cisco:Avpair="subscriber:service-name=<service-name>"

Because this command is session-oriented, it must be accompanied by one or more of the session identification attributes described in the "Session Identification, on page 6" section. If the device cannot locate a session, it returns a CoA-NAK message with the "Session Context Not Found" error-code attribute. If the device locates a session, it initiates a deactivate template operation for the hosting port and a CoA-ACK is returned. If deactivating the template fails, a CoA-NAK message is returned with the Error-Code attribute set to the appropriate message.

If the device fails before returning a CoA-ACK to the client, the process is repeated on the new active device when the request is re-sent from the client. If the device fails after returning a CoA-ACK message to the client but before the operation is complete, the operation is restarted on the new active device.
CoA Bounce Host Port Command

The CoA bounce host port command terminates a session and bounces the port (initiates a link down event followed by a link up event). The AAA server sends the request in a standard CoA-Request message with the following VSA:

Cisco:Avpair="subscriber:command=bounce-host-port"

Because this command is session-oriented, it must be accompanied by one or more of the session identification attributes described in the “Session Identification, on page 6” section. If the session cannot be located, the device returns a CoA-NAK message with the “Session Context Not Found” error-code attribute. If the session is located, the device disables the hosting port for a period of ten seconds, reenables it (port bounce), and returns a CoA-ACK.

If the device fails before returning a CoA-ACK to the client, the process is repeated on the new active device when the request is re-sent from the client. If the device fails after returning a CoA-ACK message to the client but before the operation is complete, the operation is restarted on the new active device.

The CoA bounce port command is useful as a last resort when an endpoint needs to acquire a new IP address after a change in authorization and this is the only way to indicate to the endpoint to restart the DHCP process. This can occur when there is a VLAN change and the endpoint is a device, such as a printer, that does not have a mechanism to detect a change on this authentication port. This command can cause a link flap on an authentication port, which triggers DHCP renegotiation from one or more hosts connected to this port.

CoA Disable Host Port Command

The CoA disable host port command administratively shuts down the authentication port that is hosting a session, which terminates the session. The AAA server sends the request in a standard CoA-Request message with the following VSA:

Cisco:Avpair="subscriber:command=disable-host-port"

Because this command is session-oriented, it must be accompanied by one or more of the session identification attributes described in the “Session Identification, on page 6” section. If the device cannot locate the session, it returns a CoA-NAK message with the “Session Context Not Found” error-code attribute. If the device locates the session, it disables the hosting port and returns a CoA-ACK message.

If the device fails before returning a CoA-ACK to the client, the process is repeated on the new active device when the request is re-sent from the client. If the device fails after returning a CoA-ACK message to the client but before the operation is complete, the operation is restarted on the new active device.

CoA Session Query Command

The CoA session query command requests service information about a subscriber session. The AAA server sends the request in a standard CoA-Request message containing the following VSA:

Cisco:Avpair="subscriber:command=session-query"

Because this command is session-oriented, it must be accompanied by one or more of the session identification attributes described in the “Session Identification, on page 6” section. If the device cannot locate a session, it returns a CoA-NAK message with the “Session Context Not Found” error-code attribute. If the device locates a session, it performs a session query operation on the session and returns a CoA-ACK message.
If the device fails before returning a CoA-ACK to the client, the process is repeated on the new active device when the request is re-sent from the client. If the device fails after returning a CoA-ACK message to the client but before the operation is complete, the operation is restarted on the new active device.

**CoA Session Reauthenticate Command**

To initiate session authentication, the AAA server sends a standard CoA-Request message containing the following VSAs:

- Cisco:Avpair="subscriber:command=reauthenticate"
- Cisco:Avpair="subscriber:reauthenticate-type=<last | rerun>"

"reauthenticate-type" defines whether the CoA reauthentication request uses the authentication method that last succeeded on the session or whether the authentication process is completely rerun.

The following rules apply:

- "subscriber:command=reauthenticate" must be present to trigger a reauthentication.
- If "subscriber:reauthenticate-type" is not specified, the default behavior is to rerun the last successful authentication method for the session. If the method reauthenticates successfully, all old authorization data is replaced with the new reauthenticated authorization data.
- "subscriber:reauthenticate-type" is valid only when included with "subscriber:command=reauthenticate." If it is included in another CoA command, the VSA will be silently ignored.

If the device fails before returning a CoA-ACK to the client, the process is repeated on the new active device when the request is resent from the client. If the device fails after returning a CoA-ACK message to the client but before the operation is complete, the operation is restarted on the new active device.

**CoA Session Terminate Command**

A CoA Disconnect-Request command terminates a session without disabling the host port. This command causes reinitialization of the authenticator state machine for the specified host, but does not restrict the host’s access to the network. If the session cannot be located, the device returns a Disconnect-NAK message with the "Session Context Not Found" error-code attribute. If the session is located, the device terminates the session. After the session has been completely removed, the device returns a Disconnect-ACK.

If the device fails before returning a CoA-ACK to the client, the process is repeated on the new active device when the request is re-sent from the client.

To restrict a host’s access to the network, use a CoA Request with the Cisco:Avpair="subscriber:command=disable-host-port" VSA. This command is useful when a host is known to cause problems on the network and network access needs to be immediately blocked for the host. When you want to restore network access on the port, reenable it using a non-RADIUS mechanism.
## Additional References

### Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS commands</td>
<td>Cisco IOS Master Command List, All Releases</td>
</tr>
<tr>
<td>Identity-Based Networking Services commands</td>
<td>Cisco IOS Identity-Based Networking Services Command Reference</td>
</tr>
<tr>
<td>Address Resolution Protocol (ARP) commands</td>
<td>Cisco IOS IP Addressing Services Command Reference</td>
</tr>
<tr>
<td>ARP configuration tasks</td>
<td>IP Addressing - ARP Configuration Guide</td>
</tr>
<tr>
<td>Authentication, authorization, and accounting (AAA) configuration tasks</td>
<td>Authentication Authorization and Accounting Configuration Guide</td>
</tr>
<tr>
<td>AAA commands</td>
<td>Cisco IOS Security Command Reference</td>
</tr>
</tbody>
</table>

### Standards and RFCs

<table>
<thead>
<tr>
<th>Standard/RFC</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFC 5176</td>
<td>Dynamic Authorization Extensions to RADIUS</td>
</tr>
</tbody>
</table>

### Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.</td>
<td><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></td>
</tr>
</tbody>
</table>
Feature Information for CoA Support

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

**Table 3: Feature Information for CoA Support**

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Change of Authorization</td>
<td>Cisco IOS Release 15.2(1)SY</td>
<td>Supports CoA requests for initiating the following:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Activating and deactivating service templates on sessions</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Port bounce</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Port shutdown</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Querying a session</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Re authenticating a session</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Terminating a session</td>
</tr>
<tr>
<td></td>
<td></td>
<td>These VSAs are sent in a standard CoA-Request message from a AAA server.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>In Cisco Release 15.2(1)SY, this feature is supported on Cisco Catalyst 6500 Series Switches.</td>
</tr>
</tbody>
</table>
Web Authentication Redirection to Original URL

The Web Authentication Redirection to Original URL feature enables networks to redirect guest users to the URL that they had originally requested. This feature is enabled by default and requires no configuration. This module provides information about this feature.

- Information About Web Authentication Redirection to Original URL, page 13
- Additional References for Web Authentication Redirection to Original URL, page 15
- Feature Information for Web Authentication Redirection to Original URL, page 16

Information About Web Authentication Redirection to Original URL

Web Authentication Redirection to Original URL Overview

The Web Authentication Redirection to Original URL feature enables networks to redirect guest users to the URL that they had originally requested. This feature is enabled by default and requires no configuration.

Guest networks are network connections provided by an enterprise to allow their guests to gain access to the Internet and to their own enterprise networks without compromising the security of the host enterprise. Guest users of an enterprise network can connect to the guest access network through either a wired Ethernet connection or a wireless connection.

Guest access uses a captive portal to gather all web requests made by guests and redirect these requests to one of the guest on-boarding web pages. When guests successfully complete the guest workflow, they are redirected to the page that they had originally requested.

The originally requested URL is passed as metadata along with the Cisco Identity Services Engine (ISE) guest access redirect URL. The Cisco ISE is a security policy management and control platform. It automates and simplifies access control and security compliance for wired, wireless, and VPN connectivity. The requested URL is added at the end of the Cisco ISE guest URL so that the device can send the redirect URL to the guest client. The Cisco ISE parses the URL and redirects the guest to the original URL after completing the on-boarding.
The following is an example of a redirect URL along with the original requested URL:

In this example, the URL, https://10.64.67.92:8443/guestportal/gateway?sessionId=0920269E0000000B0002426B&action=cwa is the URL for the guest portal, "&" tells the browser that what follows is a list of name value pairs, and redirect_url=http://www.cisco.com identifies the URL that the user originally requested and to which the user is redirected after completing the guest workflow.

This illustration displays the packet flow that redirects a user to the originally requested URL:

Figure 2: Original URL Redirection Packet Flow
1 A user accesses a network for the first time and sends an HTTP request to access www.google.com. When the user first accesses the network, a MAC authentication bypass (MAB) is triggered and the MAC address is sent to the Cisco ISE.

2 The Cisco ISE returns a RADIUS access-accept message (even if the MAC address is not received) along with the redirect access control list (ACL), the ACL-WEBAUTH-REDIRECT message, and the guest web portal URL to the device.

   The RADIUS message instructs the device to open a port that is restricted based on the configured port and the redirect ACLs, for regular network traffic.

3 When the user launches a web browser, the device intercepts the HTTP traffic and redirects the browser to the Cisco ISE central web authentication (CWA) guest web portal URL; the user-requested URL is extracted and appended to the Cisco ISE guest URL.

4 When the user is authenticated, the Cisco ISE sends the Device Registration page to the user. The user enters the required information, and the page is returned to the Cisco ISE. The Cisco ISE downloads user profiles and redirects the user to the originally requested URL: www.google.com.

**Additional References for Web Authentication Redirection to Original URL**

### Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS commands</td>
<td>Cisco IOS Master Command List, All Releases</td>
</tr>
<tr>
<td>IBNS commands</td>
<td>Cisco IOS Identity-Based Networking Services Command Reference</td>
</tr>
<tr>
<td>Wired guest access</td>
<td>&quot;Wired Guest Access&quot; module of the Identity-Based Networking Services Configuration Guide</td>
</tr>
</tbody>
</table>
Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support website provides extensive online resources, including</td>
<td><a href="http://www.cisco.com/support">http://www.cisco.com/support</a></td>
</tr>
<tr>
<td>documentation and tools for troubleshooting and resolving technical issues</td>
<td></td>
</tr>
<tr>
<td>with Cisco products and technologies.</td>
<td></td>
</tr>
<tr>
<td>To receive security and technical information about your products, you can</td>
<td></td>
</tr>
<tr>
<td>subscribe to various services, such as the Product Alert Tool (accessed</td>
<td></td>
</tr>
<tr>
<td>from Field Notices), the Cisco Technical Services Newsletter, and Really</td>
<td></td>
</tr>
<tr>
<td>Simple Syndication (RSS) Feeds.</td>
<td></td>
</tr>
<tr>
<td>Access to most tools on the Cisco Support website requires a Cisco.com user</td>
<td></td>
</tr>
<tr>
<td>ID and password.</td>
<td></td>
</tr>
</tbody>
</table>

Feature Information for Web Authentication Redirection to Original URL

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web Authentication Redirection to Original URL</td>
<td></td>
<td>The Web Authentication Redirection to Original URL feature enables networks to redirect guest users to the original URL that they had request. This feature is enabled by default and requires no configuration. No commands were added or updated for this feature.</td>
</tr>
</tbody>
</table>
Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About Identity Control Policies

Identity-Based Networking Services Configuration

To convert all relevant authentication commands to their Class-Based Policy Language(CPL) control policy equivalents, use the authentication convert-to new-style command. This command permanently converts the legacy configuration on the switch to identity-based networking services.
This configuration is irreversible. It disables the conversion command — `authentication display [legacy | new-style].`

Use the `authentication display config-mode` command in EXEC mode to display the current configuration mode; `legacy` if it is legacy mode and `new-style` if it is Identity-Based Networking Services configuration mode.

```
(Device)# authentication display config-mode
Current configuration mode is legacy

Device)# authentication display config-mode
Current configuration mode is new-style
```

## Concurrent Authentication Methods

Identity-Based Networking Services allows the concurrent operation of IEEE 802.1x (dot1x), MAC authentication bypass (MAB), and web authentication methods, making it possible to invoke multiple authentication methods in parallel on a single subscriber session. This allows the client-supported method to complete at the earliest opportunity without the delays associated with serialization.

Typically, the access control method that is used to authorize a host is left up to the endpoint. For example, a printer without an 802.1x supplicant would be authorized through MAB only, an employee desktop through 802.1x only, and a guest through web authentication only. The default priority order is 802.1x, followed by MAB, then web authentication. When method priorities are the same, the first method that successfully authenticates the session prevails.

An example in which more than one method may succeed during the lifetime of a session is when MAB is used to provide interim access pending success of 802.1x. A host could be also be given interim access to a web server to allow credentials to be updated so that 802.1x can succeed after an authentication failure.

## Configuration Display Mode

Identity-Based Networking Services introduces new Cisco IOS commands that replace many of the previously supported authentication and policy commands. These commands are available only after enabling the Cisco common classification policy language (C3PL) display mode that supports Identity-Based Networking Services. Identity-Based Networking Services features such as concurrent authentication and web authentication with IPv6 are not supported in legacy mode.

The device defaults to the legacy configuration mode until you do one of the following:

- Enter the `authentication display new-style` command—This command switches to C3PL display mode, temporarily converting your legacy configuration to a Identity-Based Networking Services configuration so you can see how it looks before you make the conversion permanent. You can switch back to legacy mode by using the `authentication display legacy` command. See the "Enabling the Display Mode for Identity-Based Networking Services, on page 21" section.

- Enter a Identity-Based Networking Services configuration command—After you enter the first explicit Identity-Based Networking Services command, the configuration converts to C3PL display mode permanently and legacy commands are suppressed. The `authentication display` command is disabled and you can no longer revert to the legacy configuration mode.
Control Policies for Identity-Based Networking Services

A control policy defines the handling of different subscriber life-cycle events. For various events, such as session start or session failure, you can specify actions in the control policy. These actions can be executed conditionally for different subscribers based on various match criteria. Control policies are activated on interfaces and typically control the authentication of subscriber identity and the activation of services on sessions. For example, you can configure a control policy to authenticate specific subscribers and then provide them with access to specific services.

A control policy consists of one or more control policy rules and a decision strategy that governs how the policy rules are evaluated. A control policy rule consists of a control class (a flexible condition clause), an event for which the condition is evaluated, and one or more actions. Actions are general system functions, such as "authenticate" or "activate." You define the specific actions that an event will trigger and some events have default actions.

The figure below illustrates how each control policy contains a list of events that are considered applicable to the subscriber life cycle. Within each event type is a list of control classes with different match criteria for subscriber identity, and under each class is a list of actions to be executed.

Figure 3: Control Policy Structure
Control Policy Configuration Overview

Control policies express system functionality in terms of an event, a condition, and an action. There are three steps in defining a control policy:

1. Create one or more control classes—A control class specifies the conditions that must be met for a control policy to be activated. A control class can contain multiple conditions, each of which will evaluate as either true or false. Match directives specify whether all, any, or none of the individual conditions must evaluate true for the class to evaluate true. Or, you can specify the default control class which does not contain any conditions and always evaluates true.

2. Create a control policy—A control policy contains one or more control policy rules. A control policy rule consists of a control class, an event that causes the class to be evaluated, and one or more actions. Actions are numbered and executed sequentially.

3. Apply the control policy—A control policy is activated by applying it to an interface.

Parameter Maps for Identity-Based Networking Services

A parameter map allows you to specify parameters that control the behavior of actions specified under a control policy. For Identity-Based Networking Services, an authentication parameter map defines parameters used for the action specified with the `authenticate using webauth` command. You can configure the following types of parameter maps:

- Authentication bypass (This is also called nonresponsive host [NRH] authentication.)
- Consent
- Web authentication
- Web authentication with consent

Parameter maps are optional. If you do not configure a named parameter map, the software uses the default parameters that are specified in the global parameter map.

Per User Inactivity Handling Across Methods

A common inactivity aging feature extends support for RADIUS attributes 28 (Idle-Timeout) and attribute 29 (Termination-Action) to web authenticated sessions, providing consistent inactivity handling across all authentication methods, including 802.1x, MAC authentication bypass (MAB), and web authentication. The AAA server sends these attributes as part of the user authorization. After a session has been idle for the amount of time specified in attribute 28, or has reached the timeout configured with attribute 29, the session is terminated.

You can also apply the inactivity timeout and absolute timeout to sessions through a locally defined service template. When enabling the inactivity timeout, you can also enable address resolution protocol (ARP) probes that are sent before the session is terminated. For configuration information, see the "Configuring Identity Service Templates, on page 53" module.
How to Configure Identity Control Policies

Enabling the Display Mode for Identity-Based Networking Services

Identity-Based Networking Services features are configured in the Cisco common classification policy language (C3PL) display mode. The legacy authentication manager mode is enabled by default. You can use the following procedure to switch to C3PL display mode and temporarily convert any legacy configuration commands to their C3PL equivalents. This allows you to preview your legacy configuration as a Identity-Based Networking Services configuration before making the conversion permanent. After you enter an explicit Identity-Based Networking Services command, the conversion becomes permanent and you can no longer revert to legacy mode.

**SUMMARY STEPS**

1. `enable`
2. `authentication display {legacy | new-style}`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> <code>enable</code></td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong> <code>Device&gt; enable</code></td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td><strong>Step 2</strong> `authentication display {legacy</td>
<td>new-style}`</td>
</tr>
<tr>
<td><strong>Example:</strong> <code>Device# authentication display new-style</code></td>
<td>• The default display mode is legacy.</td>
</tr>
<tr>
<td></td>
<td>• You can use this command to switch between legacy and C3PL display mode until you execute the first explicit Identity-Based Networking Services command. After you enter the first explicit Identity-Based Networking Services command, for example when configuring a control class or control policy, the system displays a prompt to confirm whether you want to continue because this command will be disabled and you cannot revert to legacy mode.</td>
</tr>
</tbody>
</table>

**Note**

If you save the configuration while the new-style mode is enabled, and then perform a reload, the display mode is permanently set to new-style. The `authentication display` command is disabled and you cannot revert to legacy mode.

If you boot the standby device while in new-style mode, the standby device will be in new-style mode and after switchover the device remains in new-style mode. To switch back to legacy mode, you must use the `authentication display legacy` command and reload the standby switch.
Configuring a Control Class

A control class defines the conditions under which the actions of a control policy are executed. You define whether all, any, or none of the conditions must evaluate true to execute the actions of the control policy. Control classes are evaluated based on the event specified in the control policy.

**Note**

This procedure shows all of the match conditions that you can configure in a control class. You must specify at least one condition in a control class to make it valid. All other conditions, and their corresponding steps, are optional (steps 4 through 18 below).

### SUMMARY STEPS

1. enable
2. configure terminal
3. class-map type control subscriber {match-all | match-any | match-none} control-class-name
4. {match | no-match} activated-service-template template-name
5. {match | no-match} authorization-status {authorized | unauthorized}
6. {match | no-match} authorizing-method-priority {eq | gt | lt} priority-value
7. {match | no-match} client-type {data | switch | video | voice}
8. {match | no-match} current-method-priority {eq | gt | lt} priority-value
9. {match | no-match} ip-address ip-address
10. {match | no-match} ipv6-address ipv6-address
11. {match | no-match} mac-address mac-address
12. {match | no-match} method {dot1x | mab | webauth}
13. {match | no-match} port-type {l2-port | l3-port | dot11-port}
14. {match | no-match} result-type [method {dot1x | mab | webauth}] result-type
15. {match | no-match} service-template template-name
16. {match | no-match} tag tag-name
17. {match | no-match} timer timer-name
18. {match | no-match} username username
19. end
20. show class-map type control subscriber {all | name control-class-name}

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong> Device&gt; enable</td>
<td></td>
</tr>
<tr>
<td>• Enter your password if prompted.</td>
<td></td>
</tr>
</tbody>
</table>

---

Identity-Based Networking Services Configuration Guide, Cisco IOS Release 15SY
<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device# configure terminal</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>class-map type control subscriber {match-all</td>
<td>match-any</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config)# class-map type control subscriber match-all DOT1X_NO_AGENT</td>
<td>• <strong>match-all</strong>—All of the conditions in the control class must evaluate true.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>match-any</strong>—At least one of the conditions in the control class must evaluate true.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>match-none</strong>—All of the conditions in the control class must evaluate false.</td>
</tr>
<tr>
<td>4</td>
<td>{match</td>
<td>no-match} activated-service-template template-name</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config-filter-control-classmap)# match activated-service-template SVC_1</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>{match</td>
<td>no-match} authorization-status {authorized</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config-filter-control-classmap)# match authorization-status authorized</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>{match</td>
<td>no-match} authorizing-method-priority {eq</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config-filter-control-classmap)# match authorizing-method-priority eq 10</td>
<td>• <strong>eq</strong>—Current priority is equal to <code>priority-value</code>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>gt</strong>—Current priority is greater than <code>priority-value</code>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>lt</strong>—Current priority is less than <code>priority-value</code>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <code>priority-value</code>—Priority value to match. Range: 1 to 254, where 1 is the highest priority and 254 is the lowest.</td>
</tr>
<tr>
<td>7</td>
<td>{match</td>
<td>no-match} client-type {data</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config-filter-control-classmap)# match client-type data</td>
<td></td>
</tr>
<tr>
<td><strong>Step</strong></td>
<td><strong>Command or Action</strong></td>
<td><strong>Purpose</strong></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
</tbody>
</table>
| **Step 8** | `{match | no-match} current-method-priority {eq | gt | lt} priority-value` | (Optional) Creates a condition that evaluates true based on the priority of the current authentication method.  
*Example:*  
Device(config-filter-control-classmap)# match current-method-priority eq 10 |
| **Step 9** | `{match | no-match} ip-address ip-address` | (Optional) Creates a condition that evaluates true based on an event’s source IPv4 address.  
*Example:*  
Device(config-filter-control-classmap)# match ip-address 10.10.10.1 |
| **Step 10** | `{match | no-match} ipv6-address ipv6-address` | (Optional) Creates a condition that evaluates true based on an event’s source IPv6 address.  
*Example:*  
Device(config-filter-control-classmap)# match ipv6-address FE80::1 |
| **Step 11** | `{match | no-match} mac-address mac-address` | (Optional) Creates a condition that evaluates true based on an event’s MAC address.  
*Example:*  
Device(config-filter-control-classmap)# match mac-address aabb.cc00.6500 |
| **Step 12** | `{match | no-match} method {dot1x | mab | webauth}` | (Optional) Creates a condition that evaluates true based on an event’s authentication method.  
*Example:*  
Device(config-filter-control-classmap)# match method dot1x |
| **Step 13** | `{match | no-match} port-type {l2-port | l3-port | dot11-port}` | (Optional) Creates a condition that evaluates true based on an event’s interface type.  
*Example:*  
Device(config-filter-control-classmap)# match port-type l2-port |
| **Step 14** | `{match | no-match} result-type [method {dot1x | mab | webauth}] result-type` | (Optional) Creates a condition that evaluates true based on the specified authentication result.  
- To display the available result types, use the question mark (?) online help function.  
*Example:*  
Device(config-filter-control-classmap)# match result-type agent-not-found |
| **Step 15** | `{match | no-match} service-template template-name` | (Optional) Creates a condition that evaluates true based on an event’s service template.  
*Example:*  
Device(config-filter-control-classmap)# match service-template svc_1 |
### Command or Action

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>`{match</td>
<td>no-match} tag tag-name`</td>
</tr>
<tr>
<td>Example:</td>
<td>Device(config-filter-control-classmap)# match tag tag_1</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>`{match</td>
<td>no-match} timer timer-name`</td>
</tr>
<tr>
<td>Example:</td>
<td>Device(config-filter-control-classmap)# match timer restart</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>`{match</td>
<td>no-match} username username`</td>
</tr>
<tr>
<td>Example:</td>
<td>Device(config-filter-control-classmap)# match username josmiths</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td><code>end</code></td>
<td>(Optional) Exits control class-map filter configuration mode and returns to privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>Device(config-filter-control-classmap)# end</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>`show class-map type control subscriber {all</td>
<td>name control-class-name}`</td>
</tr>
<tr>
<td>Example:</td>
<td>Device# show class-map type control subscriber all</td>
<td></td>
</tr>
</tbody>
</table>

#### Example: Control Class

The following example shows a control class that is configured with two match conditions:

```
class-map type control subscriber match-all DOT1X_NO_AGENT
  match method dot1x
  match result-type agent-not-found
```

### Configuring a Control Policy

Control policies determine the actions that the system takes in response to specified events and conditions. The control policy contains one or more control policy rules that associate a control class with one or more actions. The actions that you can configure in a policy rule depend on the type of event that you specify.
This task includes all of the actions that you can configure in a control policy regardless of the event. All of these actions, and their corresponding steps, are optional (steps 6 through 21 below). To display the supported actions for a particular event, use the question mark (?) online help function.

**SUMMARY STEPS**

1. `enable`
2. `configure terminal`
3. `policy-map type control subscriber control-policy-name`
4. `event event-name [match-all | match-first]`
5. `priority-number class {control-class-name | always} [do-all | do-until-failure | do-until-success]`
6. `action-number activate {policy type control subscriber control-policy-name [child [no-propagation | concurrent] | service-template template-name [aaa-list list-name] [precedence number] [replace-all]} [do-all | do-until-failure | do-until-success]`
7. `action-number authenticate using {dot1x | mab | webauth} [aaa {authc-list authc-list-name | authz-list authz-list-name}] [merge] [parameter-map map-name] [priority priority-number] [replace | replace-all] [retries number {retry-time seconds}]`
8. `action-number authentication-restart seconds`
9. `action-number authorize`
10. `action-number clear-authenticated-data-hosts-on-port`
11. `action-number clear-session`
12. `action-number deactivate {policy type control subscriber control-policy-name | service-template template-name} [child]`
13. `action-number err-disable`
14. `action-number pause reauthentication`
15. `action-number protect`
16. `action-number replace`
17. `action-number restrict`
18. `action-number resume reauthentication`
19. `action-number set-timer timer-name seconds`
20. `action-number terminate {dot1x | mab | webauth} [child]
21. `action-number unauthorize`
22. `end`
23. `show policy-map type control subscriber {all | name control-policy-name}`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong> Device&gt; enable</td>
<td></td>
</tr>
<tr>
<td><em>Enter your password if prompted.</em></td>
<td></td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>------------------</td>
<td>---------</td>
</tr>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong> Device# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> policy-map type control subscriber control-policy-name</td>
<td>Defines a control policy for subscriber sessions.</td>
</tr>
<tr>
<td><strong>Example:</strong> Device(config)# policy-map type control POLICY_1</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> event event-name [match-all</td>
<td>match-first]</td>
</tr>
<tr>
<td><strong>Example:</strong> Device(config-event-control-policymap)# event session-started</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong> priority-number class {control-class-name</td>
<td>always} [do-all</td>
</tr>
<tr>
<td><strong>Example:</strong> Device(config-class-control-policymap)# 10 class always</td>
<td></td>
</tr>
<tr>
<td><strong>Step 6</strong> action-number activate {policy type control subscriber control-policy-name [child no-propagation</td>
<td>concurrent]</td>
</tr>
<tr>
<td><strong>Example:</strong> Device(config-action-control-policymap)# 10 activate service-template FALLBACK</td>
<td></td>
</tr>
<tr>
<td><strong>Step 7</strong> action-number authenticate using {dot1x</td>
<td>mab</td>
</tr>
<tr>
<td><strong>Example:</strong> Device(config-action-control-policymap)# 10 authenticate using dot1x priority 10</td>
<td></td>
</tr>
<tr>
<td>Step</td>
<td>Command or Action</td>
</tr>
<tr>
<td>------</td>
<td>------------------</td>
</tr>
<tr>
<td>8</td>
<td>action-number authentication-restart seconds</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config-action-control-policymap)# 20 authentication-restart 60</td>
</tr>
<tr>
<td>9</td>
<td>action-number authorize</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config-action-control-policymap)# 10 authorize</td>
</tr>
<tr>
<td>10</td>
<td>action-number clear-authenticated-data-hosts-on-port</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config-action-control-policymap)# 20 clear-authenticated-data-hosts-on-port</td>
</tr>
<tr>
<td>11</td>
<td>action-number clear-session</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config-action-control-policymap)# 30 clear-session</td>
</tr>
<tr>
<td>12</td>
<td>action-number deactivate {policy type control subscriber control-policy-name</td>
</tr>
<tr>
<td></td>
<td>service-template template-name}</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config-action-control-policymap)# 20 deactivate service-template</td>
</tr>
<tr>
<td>13</td>
<td>action-number err-disable</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config-action-control-policymap)# 10 err-disable</td>
</tr>
<tr>
<td>14</td>
<td>action-number pause reauthentication</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config-action-control-policymap)# 20 pause reauthentication</td>
</tr>
<tr>
<td>15</td>
<td>action-number protect</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config-action-control-policymap)# 10 protect</td>
</tr>
<tr>
<td>16</td>
<td>action-number replace</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config-action-control-policymap)# 10 replace</td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------</td>
</tr>
</tbody>
</table>
| **Step 17** | action-number restrict  
**Example:**  
Device(config-action-control-policymap)# 10 restrict |
|  | (Optional) Drops violating packets and generates a syslog entry after a session violation event. |
| **Step 18** | action-number resume reauthentication  
**Example:**  
Device(config-action-control-policymap)# 20 resume reauthentication |
|  | (Optional) Resumes the reauthentication process after an authentication failure. |
| **Step 19** | action-number set-timer timer-name seconds  
**Example:**  
Device(config-action-control-policymap)# 20 set-timer restart 60 |
|  | (Optional) Starts a named policy timer. |
| **Step 20** | action-number terminate {dot1x | mab | webauth}  
**Example:**  
Device(config-action-control-policymap)# 20 terminate webauth |
|  | (Optional) Terminates an authentication method on a subscriber session. |
| **Step 21** | action-number unauthorize  
**Example:**  
Device(config-action-control-policymap)# 20 unauthorize |
|  | (Optional) Removes all authorization data from a subscriber session. |
| **Step 22** | end  
**Example:**  
Device(config-action-control-policymap)# end |
|  | (Optional) Exits control policy-map action configuration mode and returns to privileged EXEC mode. |
| **Step 23** | show policy-map type control subscriber {all | name control-policy-name}  
**Example:**  
Device# show policy-map type control subscriber name POLICY_1 |
|  | (Optional) Displays information about identity control policies. |

### Example: Control Policy

The following example shows a simple control policy with the minimum configuration necessary for initiating authentication:

```
policy-map type control subscriber POLICY_1  
event session-started match-all  
10 class always do-until-failure  
10 authenticate using dot1x
```
For detailed examples of control policies for concurrent and sequential authentication, see the “Configuration Examples for Identity Control Policies, on page 36” section.

Applying a Control Policy to an Interface

Control policies typically control the authentication of subscriber identity and the activation of services on sessions. Perform this task to apply a control policy to an interface.

SUMMARY STEPS

1. enable
2. configure terminal
3. interface type number
4. service-policy type control subscriber control-policy-name
5. subscriber aging {inactivity-timer seconds [probe] | probe}

DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>Example: Device&gt; enable</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td>Step 2</td>
<td>configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td></td>
<td>Example: Device# configure terminal</td>
<td></td>
</tr>
<tr>
<td>Step 3</td>
<td>interface type number</td>
<td>Specifies an interface and enters interface configuration mode.</td>
</tr>
<tr>
<td></td>
<td>Example: Device(config)# interface tengigabitethernet 1/0/1</td>
<td></td>
</tr>
<tr>
<td>Step 4</td>
<td>service-policy type control subscriber control-policy-name</td>
<td>Applies a previously configured control policy.</td>
</tr>
<tr>
<td></td>
<td>Example: Device(config-if)# service-policy type control subscriber POLICY_1</td>
<td>• To display a list of all configured control policies, use the question mark (?) online help function.</td>
</tr>
<tr>
<td>Step 5</td>
<td>subscriber aging {inactivity-timer seconds [probe]</td>
<td>probe}</td>
</tr>
<tr>
<td></td>
<td>Example: Device(config-if)# subscriber aging inactivity-timer 60 probe</td>
<td></td>
</tr>
</tbody>
</table>
Configuring Authentication Features on Ports

Perform this task to control access to a port, including the port authorization state, host access mode, preauthentication access, and the authentication direction.

**SUMMARY STEPS**

1. enable
2. configure terminal
3. interface type number
4. access-session port-control {auto | force-authorized | force-unauthorized}
5. access-session host-mode {multi-auth | multi-domain | multi-host | single-host}
6. access-session closed
7. access-session control-direction {both | in}
8. end
9. show access-session interface interface-type interface-number [details]

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example: Device&gt; enable</td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td>Step 2 configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example: Device(config)# configure terminal</td>
<td></td>
</tr>
<tr>
<td>Step 3 interface type number</td>
<td>Enters interface configuration mode for the selected interface.</td>
</tr>
<tr>
<td>Example: Device(config)# interface gigabitethernet 1/0/2</td>
<td></td>
</tr>
<tr>
<td>Step 4 access-session port-control {auto</td>
<td>force-authorized</td>
</tr>
<tr>
<td>Example: Device(config-if)# access-session port-control auto</td>
<td>• The default value is force-authorized.</td>
</tr>
<tr>
<td>Step 5 access-session host-mode {multi-auth</td>
<td>multi-domain</td>
</tr>
</tbody>
</table>
### Configuring a Parameter Map for Web-Based Authentication

A parameter map allows you to modify parameters that control the behavior of actions configured under a control policy. A parameter map for web-based authentication sets parameters that can be applied to subscriber sessions during authentication. If you do not create a parameter map, the policy uses default parameters.

Perform the following steps to define either a global or named parameter map for web-based authentication.

**Note**
The configuration commands available in the global parameter map differ from the commands available in a named parameter map.
SUMMARY STEPS

1. enable
2. configure terminal
3. parameter-map type webauth {parameter-map-name | global}
4. banner {file location:filename | text banner-text}
5. consent email
6. custom-page {failure | login [expired] | success} device location:filename
7. max-http-conns number
8. ratelimit init-state-sessions rate-limit
9. redirect {{for-login | on-failure | on-success} url | portal {ipv4 ipv4-address | ipv6 ipv6-address}}
10. timeout init-state min minutes
11. type {authbypass | consent | webauth | webconsent}
12. virtual-ip {ipv4 ipv4-address | ipv6 ipv6-address}
13. watch-list {add-item {ipv4 ipv4-address | ipv6 ipv6-address} | dynamic-expiry-timeout minutes | enabled}
14. end
15. show ip admission status [banners | custom-pages | parameter-map [parameter-map]]

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example: Device&gt; enable</td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example: Device# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> parameter-map type webauth {parameter-map-name</td>
<td>global}</td>
</tr>
<tr>
<td>Example: Device(config)# parameter-map type webauth MAP_2</td>
<td>• The specific configuration commands supported for a global parameter map defined with the global keyword differ from the commands supported for a named parameter map defined with the parameter-map-name argument.</td>
</tr>
<tr>
<td><strong>Step 4</strong> banner {file location:filename</td>
<td>text banner-text}</td>
</tr>
<tr>
<td>Example: Device(config-params-parameter-map)# banner file flash:webauth_banner.html</td>
<td></td>
</tr>
</tbody>
</table>
### Configuring a Parameter Map for Web-Based Authentication

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>consent email</td>
<td>(Optional) Requests a user's e-mail address on the web-authentication login web page.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device(config-params-parameter-map)# consent email</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• This command is supported in named parameter maps only.</td>
</tr>
<tr>
<td>6</td>
<td>custom-page {failure</td>
<td>(Optional) Displays custom authentication proxy web pages during web-based authentication.</td>
</tr>
<tr>
<td></td>
<td>login [expired]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>success} device</td>
<td></td>
</tr>
<tr>
<td></td>
<td>location:filename</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device(config-params-parameter-map)# custom-page login device flash:webauth_login.html</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device(config-params-parameter-map)# custom-page login expired device flash:webauth_expire.html</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device(config-params-parameter-map)# custom-page success device flash:webauth_success.html</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device(config-params-parameter-map)# custom-page failure device flash:webauth_fail.html</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• You must configure all four custom HTML files. If fewer than four files are configured, the internal default HTML pages will be used.</td>
</tr>
<tr>
<td>7</td>
<td>max-http-conns number</td>
<td>(Optional) Limits the number of HTTP connections for each web authentication client.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device(config-params-parameter-map)# max-http-conns 5</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>ratelimit init-state-sessions rate-limit</td>
<td>(Optional) Limits the number of web-based authentication sessions in the Init state.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device(config-params-parameter-map)# ratelimit init-state-sessions 500</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• This command is supported in the global parameter map only.</td>
</tr>
<tr>
<td>9</td>
<td>redirect {for-login</td>
<td>(Optional) Redirects users to a particular URL during web-based authentication.</td>
</tr>
<tr>
<td></td>
<td>on-failure</td>
<td></td>
</tr>
<tr>
<td></td>
<td>on-success} url</td>
<td></td>
</tr>
<tr>
<td></td>
<td>portal {ipv4 ipv4-address</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ipv6 ipv6-address}</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device(config-params-parameter-map)# redirect portal ipv6 FE80::1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device(config-params-parameter-map)# redirect on-failure <a href="http://10.10.3.34/~sample/failure.html">http://10.10.3.34/~sample/failure.html</a></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>timeout init-state min minutes</td>
<td>(Optional) Sets the Init state timeout for web-based authentication sessions.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device(config-params-parameter-map)# timeout init-state min 15</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>type {authbypass</td>
<td>(Optional) Defines the methods supported by a web-based authentication parameter map.</td>
</tr>
<tr>
<td></td>
<td>consent</td>
<td></td>
</tr>
<tr>
<td></td>
<td>webauth</td>
<td></td>
</tr>
<tr>
<td></td>
<td>webconsent}</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device(config-params-parameter-map)# type consent</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• This command is supported in named parameter maps only.</td>
</tr>
<tr>
<td>Step 12</td>
<td>virtual-ip {ipv4 ipv4-address</td>
<td>ipv6 ipv6-address}</td>
</tr>
<tr>
<td>---------</td>
<td>-------------------------------------------------</td>
<td>-------------------------------------------------------------------</td>
</tr>
<tr>
<td>Example:</td>
<td>Device(config-params-parameter-map)# virtual-ip ipv6 FE80::1</td>
<td></td>
</tr>
<tr>
<td>Step 13</td>
<td>watch-list {add-item {ipv4 ipv4-address</td>
<td>ipv6 ipv6-address}</td>
</tr>
<tr>
<td>Example:</td>
<td>Device(config-params-parameter-map)# watch-list enabled</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device(config-params-parameter-map)# watch-list dynamic-expiry-timeout 20</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device(config-params-parameter-map)# watch-list add-item ipv6 FE80::1</td>
<td></td>
</tr>
<tr>
<td>Step 14</td>
<td>end</td>
<td>(Optional) Exits parameter-map configuration mode and returns to privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>Device(config-params-parameter-map)# end</td>
<td></td>
</tr>
<tr>
<td>Step 15</td>
<td>show ip admission status [banners</td>
<td>custom-pages</td>
</tr>
<tr>
<td>Example:</td>
<td>Device# show ip admission status custom-pages</td>
<td></td>
</tr>
</tbody>
</table>

**Example: Parameter Map for Web-Based Authentication**

```plaintext
parameter-map type webauth PMAP_2
type webconsent
timeout init-state min 15
max-http-conns 5
consent email
custom-page login device flash:webauth_login.html
custom-page success device flash:webauth_success.html
custom-page failure device flash:webauth_fail.html
custom-page login expired device flash:webauth_expire.html
```

**What to Do Next**

Apply the parameter map to sessions by specifying it in the **authenticate using** command when configuring a Control Policy. See the "CONFIGURING A CONTROL POLICY, on page 25" section.
Configuration Examples for Identity Control Policies

Example: Configuring Control Policy for Concurrent Authentication Methods

The following example shows a control policy that is configured to allow concurrent authentication. All three methods (dot1x, MAB, and web authentication) are run simultaneously when a session is started. The dot1x method is set to the highest priority and web authentication has the lowest priority, which means that if multiple methods succeed, the highest priority method is honored.

If authentication fails, the session manager checks whether all methods have failed, and if so, it sets the restart timer to 60 seconds, after which it attempts to start all three methods again. On authentication success, the session manager terminates any lower priority methods; for dot1x, this is MAB and webauth; for MAB it is webauth. Lastly, if session manager detects a dot1x client (agent-found) it triggers only dot1x to run.

The class map named ALL-FAILED checks that all three methods have run to completion (result type is none until then) and that none of them was successful. In other words, all three methods have completed and failed.

```
class-map type subscriber control match-all ALL_FAILED
  no-match result-type method dot1x none
  no-match result-type method dot1x success
  no-match result-type method mab none
  no-match result-type method mab success
  no-match result-type method webauth none
  no-match result-type method webauth success

class-map type control subscriber match-all DOT1X
  match method dot1x

class-map type control subscriber match-all MAB
  match method mab

policy-map type control subscriber CONCURRENT_DOT1X_MAB_WEBAUTH
  event session-started match-all
    10 class always do-until-failure
  20 authenticate using dot1x priority 10
  30 authenticate using webauth parameter-map WEBAUTH_DEFAULT priority 30
  event authentication-failure match-first
    10 class ALL_FAILED
    20 authentication-restart 60
  event authentication-success match-all
    10 class DOT1X
    20 terminate mab
    20 terminate webauth
    20 class MAB
    20 terminate webauth
  event agent-found match-all
    10 class always do-until-failure
    10 authenticate using dot1x priority 10
```

Note: When configuring a control policy for concurrent authentication, you must include a policy rule that explicitly terminates one method after another method of a higher priority succeeds.
Example: Configuring Control Policy for Sequential Authentication Methods

The following example shows a control policy that is configured to allow sequential authentication methods using 802.1x (dot1x), MAB, and web authentication.

```
parameter-map type webauth WEBAUTH_FALLBACK
    type webauth
!
class-map type control subscriber match-all DOT1X_NO_RESP
    match method dot1x
    match result-type method dot1x agent-not-found
!
class-map type control subscriber match-all MAB_FAILED
    match method mab
    match result-type method mab authoritative
!
policy-map type control subscriber POLICY_Gi3/0/10
    event session-started match-all
        10 class always do-until-failure
        10 authenticate using dot1x priority 10
    event authentication-failure match-first
        10 class DOT1X_NO_RESP do-until-failure
        10 terminate dot1x
        20 authenticate using mab priority 20
        20 class MAB_FAILED do-until-failure
        10 terminate mab
        20 authenticate using webauth parameter-map WEBAUTH_FALLBACK priority 30
    30 class always do-until-failure
        10 terminate dot1x
        20 terminate mab
        30 terminate webauth
    40 authentication-restart 60
    event agent-found match-all
        10 class always do-until-failure
        10 terminate dot1x
        10 terminate mab
        20 terminate webauth
        30 authenticate using dot1x priority 10
```

The following example shows a control policy that is configured to allow sequential authentication methods using 802.1x and MAB. If authentication fails, a service template for VLAN is activated.

```
service-template VLAN210
    vlan 210
!
class-map type control subscriber match-all DOT1X_FAILED
    match method dot1x
    match result-type method dot1x authoritative
!
class-map type control subscriber match-all DOT1X_NO_RESP
    match method dot1x
    match result-type method dot1x agent-not-found
!
class-map type control subscriber match-all MAB_FAILED
    match method mab
    match result-type method mab authoritative
!
policy-map type control subscriber POLICY_Gi3/0/14
    event session-started match-all
        10 class always do-until-failure
        10 authenticate using dot1x retries 2 retry-time 0 priority 10
    event authentication-failure match-first
        10 class DOT1X_NO_RESP do-until-failure
        10 terminate dot1x
        20 authenticate using mab priority 20
        20 class MAB_FAILED do-until-failure
        10 terminate mab
        20 activate service-template VLAN210
    30 authorize
    30 class DOT1X_FAILED do-until-failure
        10 terminate dot1x
```
20 authenticate using mab priority 20
40 class always do-until-failure
10 terminate dot1x
20 terminate mab
30 authentication-restart 60
event agent-found match-all
10 class always do-until-failure
10 terminate mab
20 authenticate using dot1x retries 2 retry-time 0 priority 10

Example: Configuring Parameter Maps

Global Parameter Map
The following example shows the configuration of a global parameter map:

```
parameter-map type webauth global
  timeout init-state min 15
  logging enabled
  watch-list enabled
  virtual-ip ipv6 FE80::1
  redirect on-failure http://10.10.3.34/~sample/failure.html
  ratelimit init-state-sessions 500
  max-http-conns 100
  watch-list dynamic-expiry-timeout 5000
  banner file flash:webauth_banner.html
```

Named Parameter Maps for Web Authentication and Authentication Bypass (nonresponsive host [NRH])
The following example shows the configuration of two named parameter maps; one for web authentication and one for authentication bypass. This example also shows the corresponding control policy configuration.

```
parameter-map type webauth WEBAUTH_BANNER
type webauth

parameter-map type webauth WEBAUTH_NRH
type authbypass

class-map type control subscriber match-all NRH_FAIL
  match method webauth
  match current-method-priority eq 254

policy-map type control subscriber WEBAUTH_NRH
  event session-started match-all
    10 class always do-until-failure
    10 terminate webauth
    20 authenticate using webauth parameter-map WEBAUTH_NRH priority 254
```

Named Parameter Map for Web Authentication Using Custom Pages
The following example shows the configuration of a named parameter map for web authentication that defines custom pages for the login process, along with a control policy that uses the parameter map.

```
parameter-map type webauth CUSTOM_WEBAUTH
  type webauth
  custom-page login device flash:login_page.htm
  custom-page success device flash:success_page.htm
  custom-page failure device flash:fail_page.htm
  custom-page login expired device flash:expire_page.htm

policy-map type control subscriber CUSTOM_WEBAUTH
  event session-started match-all
10 class always do-until-failure
 10 authenticate using webauth parameter-map CUSTOM_WEB retries 2 retry-time 0

Named Parameter Map for Consent

The following example shows the configuration of a named parameter map for consent, along with the corresponding control policy that uses the parameter map:

parameter-map type webauth CONSENT
  type consent
  !
  ip access-list extended GUEST_ACL
  permit ip any 172.30.30.0 0.0.0.255
  permit ip any host 172.20.249.252
  !
  service-template GUEST_POLICY
  access-group GUEST_ACL
  !
  policy-map type control subscriber CONSENT
  event session-started match-all
    10 class always do-until-failure
      10 authenticate using webauth parameter-map CONSENT
  event authentication-success match-all
    10 class always do-until-failure
      10 activate service-template GUEST_POLICY

Named Parameter Map for Web Authentication with Consent

The following example shows the configuration of a named parameter map for web authentication with consent, along with the corresponding control policy that uses the parameter map:

parameter-map type webauth WEBAUTH_CONSENT
  type webconsent
  !
  ip access-list extended GUEST_ACL
  permit ip any 172.30.30.0 0.0.0.255
  permit ip any host 172.20.249.252
  !
  service-template GUEST_POLICY
  access-group GUEST_ACL
  !
  policy-map type control subscriber WEBAUTH_CONSENT
  event session-started match-all
    10 class always do-until-failure
      10 authenticate using webauth parameter-map CONSENT
  event authentication-success match-all
    10 class always do-until-failure
      10 activate service-template GUEST_POLICY

Additional References

Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS commands</td>
<td>Cisco IOS Master Command List, All Releases</td>
</tr>
<tr>
<td>Identity-Based Networking Services commands</td>
<td>Cisco IOS Identity-Based Networking Services Command Reference</td>
</tr>
<tr>
<td>Address Resolution Protocol (ARP) commands</td>
<td>Cisco IOS IP Addressing Services Command Reference</td>
</tr>
</tbody>
</table>
Feature Information for Identity Control Policies

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.
### Table 4: Feature Information for Identity Control Policies

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco Common Classification Policy Language-Based Identity Configuration</td>
<td>Cisco IOS Release 15.2(1)SY</td>
<td>Identity control policies define the actions taken in response to specified events and conditions.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>In Cisco IOS Release 15.2(1)SY, this feature is supported on the Cisco Catalyst 6500 Series Switches.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The following commands were introduced: activate (policy-map action), authenticate using, authentication display, authentication-restart, authorize, banner (parameter-map webauth), class, class-map type control subscriber, clear-authenticated-data-hosts-on-port, clear session, consent email custom-page, deactivate, err-disable, event, logging enabled (parameter-map webauth), match, max-http-conns, parameter-map type webauth, pause reauthentication, policy-map type control subscriber, protect (policy-map action), ratelimit init-state-sessions, redirect (parameter-map webauth), replace, restrict, resume reauthentication, service-policy type control subscriber, set-timer, show access-session, show class-map type control subscriber, show policy-map type control subscriber, terminate, type (parameter-map webauth), unauthorize, virtual-ip, watch-list.</td>
</tr>
<tr>
<td>Concurrent Authentication</td>
<td>Cisco IOS Release 15.2(1)SY</td>
<td>Allows concurrent operation of 802.1x, MAB and web authentication methods, making it possible to invoke multiple authentication methods in parallel on a single session.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>In Cisco IOS Release 15.2(1)SY, this feature is supported on the Cisco Catalyst 6500 Series Switches.</td>
</tr>
<tr>
<td>Per User Inactivity Handling across Methods</td>
<td>Cisco IOS Release 15.2(1)SY</td>
<td>Supports RADIUS attributes 28 (Idle-Timeout) and 29 (Termination-Action).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Cisco IOS Release 15.2(1)SY</td>
</tr>
</tbody>
</table>
CHAPTER 5

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).

- Finding Feature Information, page 43
- Restrictions for Policy Classification Engine, page 43
- Information About Policy Classification Engine, page 44
- How to Configure Policy Classification Engine, page 44
- Configuration Examples for Policy Classification Engine, page 49
- Additional References for Policy Classification Engine, page 50
- Feature Information for Policy Classification Engine, page 51

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

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, smartphones, 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 Identity-Based Networking Services

To configure policies, perform the following tasks:
1 Configure a service template.
   For more information, see the Configuring Identity Services Templates module.

2 Configure an interface template.
   For more information, see the Interface Templates module.

3 Create a parameter map.

4 Create a policy map.

5 Apply the policy on a wireless LAN (WLAN).

### Configuring a Subscriber Parameter Map

#### SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `parameter-map type subscriber attribute-to-service parameter-map-name`
4. `priority-number map device-type eq device-type oui eq MAC-address`
5. `action-number interface-template interface-template-name`
6. `end`
7. `show parameter-map type subscriber attribute-to-service parameter-map-name`

#### DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td><code>enable</code></td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>Example: &lt;br&gt;Device&gt; enable</td>
<td>&lt;ul&gt;&lt;li&gt;Enter your password if prompted.&lt;/li&gt;&lt;/ul&gt;</td>
</tr>
<tr>
<td>2.</td>
<td><code>configure terminal</code></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td></td>
<td>Example: &lt;br&gt;Device# configure terminal</td>
<td></td>
</tr>
<tr>
<td>3.</td>
<td><code>parameter-map type subscriber attribute-to-service parameter-map-name</code></td>
<td>Configures a subscriber parameter map and enters parameter-map filter configuration mode.</td>
</tr>
<tr>
<td></td>
<td>Example: &lt;br&gt;Device(config)# parameter-map type subscriber attribute-to-service param-map</td>
<td></td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
<td></td>
</tr>
<tr>
<td>-------------------</td>
<td>---------</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>Maps the priority and the Organizationally Unique Identifier (OUI) of the configured device, and enters parameter-map filter submode configuration mode.</td>
<td></td>
</tr>
<tr>
<td>priority-number <strong>map device-type eq</strong> device-type <strong>oui eq</strong> MAC-address</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong> Device(config-parameter-map-filter)# 1 map device-type eq &quot;Cisco-IP-Phone-9971&quot;oui eq 08.cc.68</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td>Maps the action number to an interface template.</td>
<td></td>
</tr>
<tr>
<td>action-number <strong>interface-template</strong> interface-template-name</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong> Device(config-parameter-map-filter-submode)# 2 interface-template IP-PHONE-INTERFACE-TEMPLATE</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Step 6</strong></td>
<td>Exits parameter-map filter submode configuration mode and returns to privileged EXEC mode.</td>
<td></td>
</tr>
<tr>
<td>end</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong> Device(config-parameter-map-filter-submode)# end</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Step 7</strong></td>
<td>Displays information about the specified parameter map.</td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong> Device# parameter-map type subscriber attribute-to-service parameter-map-name</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**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

SUMMARY STEPS

1. enable
2. configure terminal
3. policy-map type control subscriber policy-map-name
4. event identity-update \{match-all | match-first\}
5. priority-number class always \{do-all | do-until-failure | do-until-success\}
6. action-number map attribute-to-service table parameter-map-name
7. end
8. show policy-map type control subscriber policy-map-name

DETAILED STEPS

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td>Enter your password if prompted.</td>
</tr>
<tr>
<td></td>
<td>Device&gt; enable</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device# configure terminal</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>policy-map type control subscriber policy-map-name</td>
<td>Defines a control policy for subscriber sessions and enters control policy-map event configuration mode.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device(config)# policy-map type control subscriber pmap</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>event identity-update {match-all</td>
<td>match-first}</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device(config-event-control-pollicymap)# event identity-update match-all</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>priority-number class always {do-all</td>
<td>do-until-failure</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device(config-class-control-pollicymap)# 1 class always do-until-failure</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>action-number map attribute-to-service table parameter-map-name</td>
<td>Maps identity-update attribute to an autoconf template.</td>
</tr>
</tbody>
</table>
Applying a Subscriber Policy to a WLAN

**SUMMARY STEPS**

1. enable
2. configure terminal
3. wlan wlan-name wlan-ID SSID
4. service-policy type control subscriber policy-map-name
5. profiling local http
6. end

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Device&gt; enable</td>
<td>♦ Enter your password if prompted.</td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>Step 7</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>end</td>
<td></td>
<td>Exits control policy-map action configuration mode and returns to privileged EXEC mode.</td>
</tr>
<tr>
<td>Example: Device(config-action-control-policymap)# end</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 8</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show policy-map type control subscriber policy-map-name</td>
<td>Displays information and statistics about the control policies.</td>
<td></td>
</tr>
<tr>
<td>Example: Device# show policy-map type control subscriber pmap</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

**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
```
<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong> Device# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> wlan wlan-name wlan-ID SSID</td>
<td>Configures a wireless LAN (WLAN) network and enters WLAN configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong> Device(config)# wlan wlan1 9 policywlan</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> service-policy type control subscriber policy-map-name</td>
<td>Defines a service policy for subscriber sessions.</td>
</tr>
<tr>
<td><strong>Example:</strong> Device(config-wlan)# service-policy type control subscriber pmap</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong> profiling local http</td>
<td>Configures client profiling on a WLAN based on HTTP attributes.</td>
</tr>
<tr>
<td><strong>Example:</strong> Device(config-wlan)# profiling local http</td>
<td></td>
</tr>
<tr>
<td><strong>Step 6</strong> end</td>
<td>Exits WLAN configuration mode and returns to privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong> Device(config-wlan)# end</td>
<td></td>
</tr>
</tbody>
</table>

## 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-service table param-map  
Device(config-action-control-policymap)# end
Example: Applying a Subscriber Policy to a WLAN

```plaintext
Device# configure terminal
Device(config)# wlan wlan1 9 policywlan
Device(config-wlan)# service-policy type control subscriber pmap
Device(config-wlan)# profiling local http
Device(config-wlan)# end
```

Additional References for Policy Classification Engine

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS commands</td>
<td>Cisco IOS Master Command List, All Releases</td>
</tr>
<tr>
<td>Identity commands</td>
<td>Cisco IOS Identity-Based Networking Services Command Reference</td>
</tr>
<tr>
<td>Service templates</td>
<td>“Configuring Identity Service Templates” module of the Identity-Based Networking Services Configuration Guide</td>
</tr>
<tr>
<td>Interface templates</td>
<td>“Interface Templates” module of the Identity-Based Networking Services Configuration Guide</td>
</tr>
</tbody>
</table>

Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</td>
<td><a href="http://www.cisco.com/support">http://www.cisco.com/support</a></td>
</tr>
</tbody>
</table>
Feature Information for Policy Classification Engine

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 5: Feature Information for Policy Classification Engine

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Policy Classification Engine</td>
<td>Cisco IOS XE Release 3.6E</td>
<td>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.</td>
</tr>
<tr>
<td></td>
<td>Cisco IOS 15.2(1)SY</td>
<td>In Cisco IOS XE 3.6E, this feature is supported on the following platforms:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Cisco 5700 Series Wireless LAN Controllers</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Cisco Catalyst 3650 Series Switches</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Cisco Catalyst 3850 Series Switches</td>
</tr>
<tr>
<td></td>
<td></td>
<td>In Cisco IOS 15.2(SY), this feature was supported on Cisco Catalyst 6500 Series Switches.</td>
</tr>
</tbody>
</table>
Configuring Identity Service Templates

Identity service templates contain a set of policy attributes or features that can be applied to one or more subscriber sessions through a control policy, a RADIUS Change of Authorization (CoA) request, or a user profile or service profile. This module provides information about how to configure local service templates for Identity-Based Networking Services.

- Finding Feature Information, page 53
- Prerequisites for Identity Service Templates, page 53
- Information About Identity Service Templates, page 54
- How to Configure Identity Service Templates, page 55
- Configuration Examples for Identity Service Templates, page 57
- Additional References, page 59
- Feature Information for Identity Service Templates, page 60

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Identity Service Templates

For downloadable service templates, the switch uses the default password "cisco123" when downloading the service templates from the authentication, authorization, and accounting (AAA) server, Cisco Secure Access Control Server (ACS), or Cisco Identity Services Engine (ISE). The AAA, ACS, and ISE server must include the password "cisco123" in the service template configuration.
Information About Identity Service Templates

Service Templates for Identity-Based Networking Services

A service template contains a set of service-related attributes or features, such as access control lists (ACLs) and VLAN assignments, that can be activated on one or more subscriber sessions in response to session life-cycle events. Templates simplify the provisioning and maintenance of network session policies where policies fall into distinct groups or are role-based.

A service template is applied to sessions through its reference in a control policy, through RADIUS Change of Authorization (CoA) requests, or through a user profile or service profile. User profiles are defined per subscriber; service profiles can apply to multiple subscribers.

Identity-Based Networking Services supports two types of service templates:

- **Downloadable Service Templates**—The service template is configured centrally on an external ACS or AAA server and downloaded on demand.
- **Locally Configured Service Templates**—The service template is configured locally on the device through the Cisco IOS command-line interface (CLI).

Downloadable Service Templates

Identity-Based Networking Services can download a service template defined on an external AAA server. The template defines a collection of AAA attributes. These templates are applied to sessions through the use of vendor-specific attributes (VSAs) included in RADIUS CoA messages received from the external AAA server or ACS. The name of the template is referenced in a user profile or a control policy, which triggers a download of the service template during processing.

The downloadable template is cached on the device and subsequent requests for a download will refer to the available cached template. The template however is cached only for the duration of its active usage. The downloaded template cached on the device is protected and cannot be deleted through the command line interface or through other applications. This ensures that the template is deleted only when there are no active references to it.

Locally Configured Service Templates

Service templates can be configured locally through the CLI. These service templates can be applied to subscriber sessions by a reference in a control policy.

When an active local template is updated, changes to that local template will be reflected across all sessions for which the template is active. If a template is deleted, all content from that template that is applied against sessions is removed.
How to Configure Identity Service Templates

Configuring a Local Service Template

A service template defines the local policies that can be applied to a subscriber session. Activate this service template on sessions on which the local policies must be applied.

**SUMMARY STEPS**

1. `enable`
2. `configure terminal`
3. `service-template template-name`
4. `absolute-timer minutes`
5. `access-group access-list-name`
6. `description description`
7. `inactivity-timer minutes [probe]`
8. `redirect url url`
9. `sgt range`
10. `tag tag-name`
11. `vlan vlan-id`
12. `sgt sgt-tag`
13. `end`
14. `show service-template [template-name]`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example: Device&gt; enable</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> <code>configure terminal</code></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example: Device# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> <code>service-template template-name</code></td>
<td>Creates a service template and enters service template configuration mode.</td>
</tr>
<tr>
<td>Example: Device(config)# service-template SVC_2</td>
<td></td>
</tr>
<tr>
<td>Step 4</td>
<td><strong>Command or Action</strong></td>
</tr>
<tr>
<td>--------</td>
<td>----------------------</td>
</tr>
<tr>
<td></td>
<td><strong>absolute-timer</strong> <em>minutes</em></td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config-service-template)# absolute-timer 15</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 5</th>
<th><strong>Command or Action</strong></th>
<th><strong>Purpose</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>access-group</strong> <em>access-list-name</em></td>
<td>(Optional) Applies an access list to sessions using a service template.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config-service-template)# access-group ACL_2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 6</th>
<th><strong>Command or Action</strong></th>
<th><strong>Purpose</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>description</strong> <em>description</em></td>
<td>(Optional) Adds a description for a service template.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config-service-template)# description label for SVC_2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 7</th>
<th><strong>Command or Action</strong></th>
<th><strong>Purpose</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>inactivity-timer</strong> <em>minutes</em></td>
<td>(Optional) Enables an inactivity timeout for subscriber sessions.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config-service-template)# inactivity-timer 15</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 8</th>
<th><strong>Command or Action</strong></th>
<th><strong>Purpose</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>redirect-url</strong> <em>url</em></td>
<td>(Optional) Redirects clients to a particular URL.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config-service-template)# redirect-url <a href="http://www.cisco.com">www.cisco.com</a></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 9</th>
<th><strong>Command or Action</strong></th>
<th><strong>Purpose</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>sgt</strong> <em>range</em></td>
<td>(Optional) Associates a Security Group Tag (SGT) with a service template.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config-service-template)# sgt 100</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 10</th>
<th><strong>Command or Action</strong></th>
<th><strong>Purpose</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>tag</strong> <em>tag-name</em></td>
<td>(Optional) Associates a user-defined tag with a service template.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config-service-template)# tag TAG_2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 11</th>
<th><strong>Command or Action</strong></th>
<th><strong>Purpose</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>vlan</strong> <em>vlan-id</em></td>
<td>(Optional) Applies a VLAN to sessions using a service template.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config-service-template)# vlan 215</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 12</th>
<th><strong>Command or Action</strong></th>
<th><strong>Purpose</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>sgt</strong> <em>sgt-tag</em></td>
<td>(Optional) Adds a Security Group Tag (SGT) using a service template.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> Device(config-service-template)# sgt</td>
<td></td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
<td></td>
</tr>
<tr>
<td>-------------------</td>
<td>---------</td>
<td></td>
</tr>
<tr>
<td><strong>Step 13</strong> end</td>
<td>Exits service template configuration mode and returns to privileged EXEC mode.</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td>Device(config-service-template)# end</td>
<td></td>
</tr>
<tr>
<td><strong>Step 14</strong> show service-template [template-name]</td>
<td>Displays information about configured service templates.</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td>Device# show service-template SVC_2</td>
<td></td>
</tr>
</tbody>
</table>

**Example: Service Template**
```
service-template SVC_2  
  description label for SVC_2  
  access-group ACL_2  
  redirect url www.cisco.com  
  vlan 215  
  inactivity-timer 15  
  absolute-timer 15  
  tag TAG_2  
```

**What to Do Next**
To activate a service template on a subscriber session, specify the service template in a control policy. See "Configuring a Control Policy, on page 25."

**Configuration Examples for Identity Service Templates**

**Example: Activating a Service Template and Replace All**

**Local Service Template Configuration**
The following example shows the configuration of a service template defined locally on the device. This template contains attributes that are applied to sessions that use the control policy named POSTURE_VALIDATION, shown below:
```
service-template DOT1X  
  access-group SVC1_ACL  
  redirect url www.cisco.com match URL_REDIRECT_ACL  
  inactivity-timer 60  
  absolute-timer 300  
  !  
  ip access-list extended URL_REDIRECT_ACL  
  permit tcp any host 5.5.5.5 eq www  
```
Control Policy Configuration

The following example shows a control policy that activates the service template named DOT1X with replace-all enabled. The successfully activated template will replace the existing authorization data and any service template previously applied to the session.

```plaintext
policy-map type control subscriber POSTURE_VALIDATION
  event session-started match-all
    10 class always do-until-failure
      10 authenticate using dot1x priority 10
      20 authenticate using webauth priority 20
  event authentication-success match-all
    10 class DOT1X do-all
    10 terminate webauth
    20 activate service-template DOT1X replace-all
```

Example: Activating a Service Template for Fallback Service

Local Service Template Configuration

The following example shows the configuration of a service template defined locally on the device. This template contains attributes that are applied to sessions that use the control policy named POSTURE_VALIDATION, shown below:

```plaintext
service-template FALLBACK
  description fallback service
  access-group ACL_2
  redirect url www.cisco.com
  inactivity-timer 15
  absolute-timer 15
  tag TAG_2
```

Control Policy Configuration

The following example shows a control policy that runs authentication methods dot1x and MAB. If dot1x authentication fails, MAB authentication is attempted. If MAB fails, the system provides a default authorization profile using the FALLBACK template.

```plaintext
policy-map type control subscriber POSTURE_VALIDATION
  event session-started match-all
    10 class always do-all
    10 authenticate using dot1x
  event authentication-failure match-all
    10 class DOT1X do-all
    10 authenticate using mab
    20 class MAB do-all
    10 activate service-template FALLBACK
```

Example: Deactivating a Service Template

Access Control List Configuration

The following example shows the configuration of an access control list (ACL) that is used by the local service template named LOW_IMPACT_TEMPLATE, shown below.

```plaintext
ip access-list extended LOW_IMPACT_ACL
  permit udp any any eq bootps
  permit tcp any any eq www
  permit tcp any any eq 443
  permit ip any 172.30.0.0 0.0.255.255
```
Local Service Template Configuration

The following example shows the configuration of the local service template that provides limited access to all hosts even when authentication fails.

```
service-template LOW_IMPACT_TEMPLATE
description Service template for Low impact mode
access-group LOW_IMPACT_ACL
inactivity-timer 60
tag LOW_IMPACT_TEMPLATE
```

Control Policy Configuration

The following example shows the configuration of a control policy that uses the template named LOW_IMPACT_TEMPLATE to provide limited access to all hosts even when authentication fails. If authentication succeeds, the policy manager removes the service template and provides access based on the policies downloaded by the RADIUS server.

```
class-map type control subscriber match-all DOT1X_MAB_FAILED
  no-match result-type method dot1x success
  no-match result-type method mab success
!
policy-map type control subscriber CONCURRENT_DOT1X_MAB_LOW_IMP_MODE
  event session-started match-all
    10 class always do-until-failure
    10 authorize
    20 activate service-template LOW_IMPACT_TEMPLATE
    30 authenticate using mab
    40 authenticate using dot1x
  event authentication-success match-all
    10 class always do-until-failure
    10 authorize
    20 activate service-template LOW_IMPACT_TEMPLATE
    30 authenticate using mab
  event authentication-failure match-first
    10 class DOT1X_MAB_FAILED do-until-failure
    10 authorize
    20 terminate dot1x
    30 terminate mab
  event agent-found match-all
    10 class always do-until-failure
    10 authenticate using dot1x
  event inactivity-timeout match-all
    10 class always do-until-failure
    10 clear-session
```

Additional References

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS commands</td>
<td>Cisco IOS Master Command List, All Releases</td>
</tr>
<tr>
<td>Identity-Based Networking Services commands</td>
<td>Cisco IOS Identity-Based Networking Services Command Reference</td>
</tr>
<tr>
<td>Address Resolution Protocol (ARP) commands</td>
<td>Cisco IOS IP Addressing Services Command Reference</td>
</tr>
<tr>
<td>ARP configuration tasks</td>
<td>IP Addressing - ARP Configuration Guide</td>
</tr>
</tbody>
</table>
Feature Information for Identity Service Templates

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 6: Feature Information for Identity Service Templates

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Cisco IOS 15.2(1)SY</td>
<td>In Cisco IOS 15.2(1)SY, this feature was supported on Cisco Catalyst 6500 Series Switches.</td>
</tr>
<tr>
<td>Feature Name</td>
<td>Releases</td>
<td>Feature Information</td>
</tr>
<tr>
<td>----------------------------</td>
<td>------------</td>
<td>-------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Identity Service Template</td>
<td>Cisco IOS 15.2(1)SY</td>
<td>Enables identity service templates to be configured locally and available at all times. In Cisco IOS 15.2(1)SY, this feature was supported on Cisco Catalyst 6500 Series Switches. The following commands were introduced: absolute-timer, access-group (service template), description (service template), inactivity-timer, redirect url, service-template, show service-template, tag (service template), vlan (service template).</td>
</tr>
</tbody>
</table>
Interface Templates

An interface template provides a mechanism to configure multiple commands at the same time and associate it with a target such as an interface. An interface template is a container of configurations or policies that can be applied to specific ports.

- Finding Feature Information, page 63
- Restrictions for Interface Templates, page 63
- Information About Interface Templates, page 64
- How to Configure Interface Templates, page 67
- Configuration Examples for Interface Templates, page 77
- Additional References for Interface Templates, page 78
- Feature Information for Interface Templates, page 78

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions for Interface Templates

- Interface templates are not applicable for wireless sessions.
- Remote storing and downloading of templates is not supported.
- Port channel configuration through template is not supported on Cisco Catalyst 6500 Series Switches.
Information About Interface Templates

About Interface Templates

An interface template is a container of configurations or policies that can be applied to specific ports. When an interface template is applied to an access port, it impacts all traffic that is exchanged on the port.

There are two types of interface templates; user and builtin templates. Builtin templates are created by the system.

You can modify builtin templates. If you delete a modified builtin template the system restores the original definition of the template.

The following are the available builtin templates:

- AP_INTERFACE_TEMPLATE (Access Point)
- DMP_INTERFACE_TEMPLATE (Digital Media Player)
- IP_CAMERA_INTERFACE_TEMPLATE
- IP_PHONE_INTERFACE_TEMPLATE
- LAP_INTERFACE_TEMPLATE (Lightweight Access Point)
- MSP_CAMERA_INTERFACE_TEMPLATE
- MSP_VC_INTERFACE_TEMPLATE (Video Conferencing)
- PRINTER_INTERFACE_TEMPLATE
- ROUTER_INTERFACE_TEMPLATE
- SWITCH_INTERFACE_TEMPLATE
- TP_INTERFACE_TEMPLATE (TelePresence)

Following is an example of a builtin interface template:

Template Name : IP_CAMERA_INTERFACE_TEMPLATE
Modified : No
Template Definition :
    spanning-tree portfast
    spanning-tree bpduguard enable
    switchport mode access
    switchport block unicast
    switchport port-security
    mls qos trust dscp
    srr-queue bandwidth share 1 30 35 5
    priority-queue out

You can also create specific user templates with the commands that you want to include.

Note
The template name must not contain spaces.

You can create an interface template using the template command in global configuration mode. In template configuration mode, enter the required commands. The following commands can be entered in template configuration mode:
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>access-session</td>
<td>Configures access session specific interface commands.</td>
</tr>
<tr>
<td>authentication</td>
<td>Configures authentication manager Interface Configuration commands.</td>
</tr>
<tr>
<td>carrier-delay</td>
<td>Configures delay for interface transitions.</td>
</tr>
<tr>
<td>dampening</td>
<td>Enables event dampening.</td>
</tr>
<tr>
<td>default</td>
<td>Sets a command to its defaults.</td>
</tr>
<tr>
<td>description</td>
<td>Configures interface-specific description.</td>
</tr>
<tr>
<td>dot1x</td>
<td>Configures interface configuration commands for IEEE 802.1X.</td>
</tr>
<tr>
<td>hold-queue</td>
<td>Sets hold queue depth.</td>
</tr>
<tr>
<td>ip</td>
<td>Configures IP template.</td>
</tr>
<tr>
<td>keepalive</td>
<td>Enables keepalive.</td>
</tr>
<tr>
<td>load-interval</td>
<td>Specifies interval for load calculation for an interface.</td>
</tr>
<tr>
<td>mab</td>
<td>Configures MAC authentication bypass Interface.</td>
</tr>
<tr>
<td>mls</td>
<td>Enables multilayer switching configurations. This command is available on</td>
</tr>
<tr>
<td></td>
<td>the following devices in template configuration mode:</td>
</tr>
<tr>
<td></td>
<td>• Cisco Catalyst 2960-S Series Switches</td>
</tr>
<tr>
<td></td>
<td>• Cisco Catalyst 2960-X Series Switches</td>
</tr>
<tr>
<td></td>
<td>• Cisco Industrial Ethernet 3000 Series Switches</td>
</tr>
<tr>
<td>peer</td>
<td>Configures peer parameters for point to point interfaces.</td>
</tr>
<tr>
<td>priority-queue</td>
<td>To set the priority-queue size for a template. This command is available</td>
</tr>
<tr>
<td></td>
<td>on the following devices in template configuration mode:</td>
</tr>
<tr>
<td></td>
<td>• Cisco Catalyst 2960-S Series Switches</td>
</tr>
<tr>
<td></td>
<td>• Cisco Catalyst 2960-X Series Switches</td>
</tr>
<tr>
<td></td>
<td>• Cisco Industrial Ethernet 3000 Series Switches</td>
</tr>
</tbody>
</table>
### Description

**queue-set**

Configures the QoS queue set on a template. This command is available on the following devices in template configuration mode:

- Cisco Catalyst 2960-S Series Switches
- Cisco Catalyst 2960-X Series Switches
- Cisco Industrial Ethernet 3000 Series Switches

**radius-server**

Enables RADIUS server configurations. This command is available on the following devices in template configuration mode:

- Catalyst 4500E Supervisor Engine 7-E
- Catalyst 4500E Supervisor Engine 7L-E
- Catalyst 4500E Supervisor Engine 8-E
- Catalyst 4500-X Series Switches

**service-policy**

Configures CPL service policy.

**source**

Gets configurations from another source.

**spanning-tree**

Configures spanning tree subsystem

**storm-control**

Configures storm control.

**subscriber**

Configures subscriber inactivity timeout value.

**switchport**

Sets switching mode configurations

**trust**

Sets trust value for the interface.

---

**Note**

System built-in templates are not displayed in the running configuration. These templates show up in the running configuration only if you edit them.

**Note**

On Cisco Catalyst 6500 Series Switches, the stateful switchover fails if `access-session` and `switchport mode access` are both configured in an interface template. To avoid the switchover failure, configure the `switchport mode access` command on the interface, instead of in an interface template.
Binding an Interface Template to a Target

Each template can be bound to a target. Template binding or sourcing can be either static or dynamic. Static binding of a template involves binding the template to a target, like an interface. Only one template can be bound at a time using static binding. Static binding of another template to the same target will unbind the previously bound template. To configure static binding, use the `source template` command in interface configuration mode.

Any number of templates can be bound dynamically to a target. To configure dynamic binding using builtin policy maps and parameter maps, enable the `autoconf enable` feature using the `autoconf enable` command.

**Note**

You can have statically and dynamically bind templates on the same interface at a time.

Priority for Configurations Using Interface Templates

Configuration applied through dynamically-bound templates has the highest priority, followed by configuration applied directly on the interface, and then configuration applied through statically-bound templates. When similar commands are present at different priority levels, the one at the highest priority is applied. If a configuration at a higher priority level is not applied, then the configuration with the next highest priority is applied to the target.

Multiple templates can be dynamically bound to a target. When multiple templates are dynamically bound, the template that is applied last has the highest priority.

To delete a template, you must remove the binding to all targets. If you bind a template that does not exist, a new template is created with no configurations.

How to Configure Interface Templates

Configuring Interface Templates

Perform the following task to create user interface templates:

**SUMMARY STEPS**

1. `enable`
2. `configure terminal`
3. `template name`
4. `load-interval interval`
5. `description description`
6. `keepalive number`
7. `end`
## DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>enable</td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td>Example:</td>
<td>Device&gt; enable</td>
</tr>
</tbody>
</table>

| **Step 2**        | Enters global configuration mode. |
| configure terminal|                                 |
| Example:          | Device# configure terminal |

| **Step 3**        | Creates a user template and enters template configuration mode. |
| template name     | Note: Builtin template are system-generated. |
| Example:          | Device(config)# template user-template1 |

| **Step 4**        | Configures the sampling interval for statistics collections on the template. |
| load-interval interval| Note: Builtin template are system-generated. |
| Example:          | Device(config-template)# load-interval 60 |

| **Step 5**        | Configures the description for the template. |
| description description | |
| Example:          | Device(config-template)# description This is a user template |

| **Step 6**        | Configures the keepalive timer. |
| keepalive number  | |
| Example:          | Device(config-template)# Keepalive 60 |

| **Step 7**        | Exits global configuration mode and returns to privileged EXEC mode. |
| end               | |
| Example:          | Device(config)# end |
Configuring Static Binding for Interface Templates

SUMMARY STEPS

1. enable
2. configure terminal
3. interface type number
4. source template name
5. end

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device&gt; enable</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> interface type number</td>
<td>Specifies the interface type and number and enters interface configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device(config)# interface GigabitEthernet 1/0/12</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> source template name</td>
<td>Statically applies an interface template to a target.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device(config-if)# source template user-template1</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong> end</td>
<td>Exits interface configuration mode and returns to privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device(config-if)# end</td>
<td></td>
</tr>
</tbody>
</table>
Example
To verify static binding use the `show running-config interface int-name` and the `show derived-config interface int-name` commands.

```
Device# show running-config interface GigabitEthernet 1/0/12
Building configuration...
Current configuration : 71 bytes
!
interface GigabitEthernet1/0/12
source template user-template1
end
Device# show derived-config interface GigabitEthernet 1/0/12
Building configuration...

Derived configuration : 108 bytes
!
interface GigabitEthernet1/0/12
description This is a user template
load-interval 60
keepalive 60
end
```

Configuring Dynamic Binding of Interface Templates

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface type number**
4. **service-policy type control subscriber policymap-name**
5. **end**

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td><code>enable</code></td>
<td>- Enter your password if prompted.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>Device&gt; enable</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><code>configure terminal</code></td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>Device# configure terminal</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>Specifies the interface type and number and enters interface configuration mode.</td>
</tr>
<tr>
<td><code>interface type number</code></td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>Device(config)# interface GigabitEthernet 4/0/1</code></td>
<td></td>
</tr>
</tbody>
</table>
### Command or Action

<table>
<thead>
<tr>
<th>Step 4</th>
<th><code>service-policy type control subscriber policymap-name</code></th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Dynamically applies an interface template to a target.</td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Device(config-if)# service-policy type control subscriber POLICY-Gi1/0/12</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 5</th>
<th><code>end</code></th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Exits interface configuration mode and returns to privileged EXEC mode.</td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Device(config-if)# end</td>
<td></td>
</tr>
</tbody>
</table>

## Verifying an Interface Template

Use one or more of the commands listed below to verify the interface template configuration.

### SUMMARY STEPS

1. `enable`
2. `show template interface all {all | binding {temp-name | all | target int-name} | brief}`
3. `show template interface source {built-in [original] | user} {temp-name | all}`
4. `show template service {all | binding target int-name | brief | source {aaa | built-in | user} {temp-name | all}}`

### DETAILED STEPS

**Step 1**  
`enable`  
**Example:**  
Device> enable  
Enables privileged EXEC mode.  
- Enter your password if prompted.

**Step 2**  
`show template interface all {all | binding {temp-name | all | target int-name} | brief}`  
Shows all interface template configurations.

**Step 3**  
`show template interface source {built-in [original] | user} {temp-name | all}`  
Shows interface template source configurations.

**Step 4**  
`show template service {all | binding target int-name | brief | source {aaa | built-in | user} {temp-name | all}}`  
Shows all interface template service configurations.
Verifying Interface User Templates

Device# show template interface source user all
Template Name : TEST-1
Template Definition:
  load-interval 60
description TEST_1_TEMPLATE
  keepalive 200
!
Template Name : TEST-2
Template Definition:
  load-interval 60
description TEST-1_TEMPLATE
  keepalive 200

Verifying all Builtin Templates

Device# show template interface source built-in all

Building configuration...

Template Name : AP_INTERFACE_TEMPLATE
Modified : No
Template Definition :
  switchport mode trunk
  switchport nonegotiate
  service-policy input AutoConf-4.0-Trust-Cos-Input-Policy
  service-policy output AutoConf-4.0-Output-Policy
!
Template Name : DMP_INTERFACE_TEMPLATE
Modified : No
Template Definition :
  switchport mode access
  switchport block unicast
  switchport port-security
  spanning-tree portfast
  spanning-tree bpduguard enable
  service-policy input AutoConf-4.0-Trust-Dscp-Input-Policy
  service-policy output AutoConf-4.0-Output-Policy
!
Template Name : IP_CAMERA_INTERFACE_TEMPLATE
Modified : No
Template Definition :
  switchport mode access
  switchport block unicast
  switchport port-security
  spanning-tree portfast
  spanning-tree bpduguard enable
  service-policy input AutoConf-4.0-Trust-Dscp-Input-Policy
  service-policy output AutoConf-4.0-Output-Policy
!
Template Name : IP_PHONE_INTERFACE_TEMPLATE
Modified : No
Template Definition :
  switchport mode access
  switchport block unicast
  switchport port-security maximum 3
  switchport port-security maximum 2 vlan access
  switchport port-security violation restrict
  switchport port-security aging time 2
  switchport port-security aging type inactivity
  switchport port-security
  storm-control broadcast level pps 1k
  storm-control multicast level pps 2k
  storm-control action trap
  spanning-tree portfast
  spanning-tree bpduguard enable
Interface Templates

Verifying an Interface Template

Template Name: LAP_INTERFACE_TEMPLATE
Modified: No
Template Definition:
switchport mode access
switchport block unicast
switchport port-security violation protect
switchport port-security aging time 2
switchport port-security aging type inactivity
switchport port-security
storm-control broadcast level pps 1k
storm-control multicast level pps 2k
storm-control action trap
spanning-tree portfast
spanning-tree bpduguard enable
ip dhcp snooping limit rate 15
load-interval 30

Template Name: MSP_CAMERA_INTERFACETEMPLATE
Modified: No
Template Definition:
switchport mode access
switchport block unicast
switchport port-security
spanning-tree portfast
spanning-tree bpduguard enable

Template Name: MSP_VC_INTERFACE_TEMPLATE
Modified: No
Template Definition:
switchport mode access
switchport port-security
spanning-tree portfast
spanning-tree bpduguard enable
load-interval 30

Template Name: PRINTER_INTERFACE_TEMPLATE
Modified: No
Template Definition:
switchport mode access
switchport port-security maximum 2
switchport port-security
spanning-tree portfast
spanning-tree bpduguard enable
load-interval 60

Template Name: ROUTER_INTERFACE_TEMPLATE
Modified: No
Template Definition:
switchport mode trunk
spanning-tree portfast trunk
spanning-tree bpduguard enable
service-policy input AutoConf-4.0-Trust-Cos-Input-Policy
service-policy output AutoConf-4.0-Output-Policy

Template Name: SWITCH_INTERFACE_TEMPLATE
Modified: No
Template Definition:
switchport mode trunk
service-policy input AutoConf-4.0-Trust-Cos-Input-Policy
service-policy output AutoConf-4.0-Output-Policy

Template Name: TP_INTERFACE_TEMPLATE
Modified: No
Template Definition:
switchport mode access
switchport port-security maximum 3
switchport port-security maximum 2 vlan access
switchport port-security violation restrict
Verifying an Interface Template

Verifying all Built-in Templates on Cisco Catalyst 2960-S Series Switches, Cisco Catalyst 2960-X Series Switches, Cisco Industrial Ethernet 3000 Series Switches

Device# show template interface source built-in all

Building configuration...

Template Name : AP_INTERFACE_TEMPLATE
Modified : No
Template Definition :
  switchport mode trunk
  switchport nonegotiate
  mls qos trust cos
  srr-queue bandwidth share 1 30 35 5
  priority-queue out

Template Definition :
  switchport mode access
  switchport block unicast
  switchport port-security
  mls qos trust dscp
  srr-queue bandwidth share 1 30 35 5
  priority-queue out

Template Name : DMP_INTERFACE_TEMPLATE
Modified : No
Template Definition :
  spanning-tree portfast
  spanning-tree bpduguard enable
  switchport mode access
  switchport block unicast
  switchport port-security
  mls qos trust dscp
  srr-queue bandwidth share 1 30 35 5
  priority-queue out

Template Name : IP_CAMERA_INTERFACE_TEMPLATE
Modified : No
Template Definition :
  spanning-tree portfast
  spanning-tree bpduguard enable
  switchport mode access
  switchport block unicast
  switchport port-security
  mls qos trust dscp
  srr-queue bandwidth share 1 30 35 5
  priority-queue out

Template Name : IP_PHONE_INTERFACE_TEMPLATE
Modified : No
Template Definition :
  spanning-tree portfast
  spanning-tree bpduguard enable
  switchport mode access
  switchport block unicast
  switchport port-security maximum 3
  switchport port-security maximum 2 vlan access
  switchport port-security violation restrict
  switchport port-security aging time 2
  switchport port-security aging type inactivity
  switchport port-security
  storm-control broadcast level pps 1k
  storm-control multicast level pps 2k
  storm-control action trap
  mls qos trust cos

Identity-Based Networking Services Configuration Guide, Cisco IOS Release 15SY

74
service-policy input AUTOCONF-SRND4-CISCOPHONE-POLICY
ip dhcp snooping limit rate 15
load-interval 30
srr-queue bandwidth share 1 30 35 5
priority-queue out
!
Template Name : LAP_INTERFACE_TEMPLATE
Modified : No
Template Definition :
  spanning-tree portfast
  spanning-tree bpduguard enable
  switchport mode access
  switchport block unicast
  switchport port-security violation protect
  switchport port-security aging time 2
  switchport port-security aging type inactivity
  switchport port-security
  storm-control broadcast level pps 1k
  storm-control multicast level pps 2k
  storm-control action trap
  mls qos trust dscp
  ip dhcp snooping limit rate 15
  load-interval 30
  srr-queue bandwidth share 10 10 60 20
  priority-queue out
!
Template Name : MSP_CAMERA_INTERFACE_TEMPLATE
Modified : No
Template Definition :
  spanning-tree portfast
  spanning-tree bpduguard enable
  switchport mode access
  switchport block unicast
  switchport port-security
!
Template Name : MSP_VC_INTERFACE_TEMPLATE
Modified : No
Template Definition :
  spanning-tree portfast
  spanning-tree bpduguard enable
  switchport mode access
  switchport block unicast
  switchport port-security violation restrict
  switchport port-security aging time 2
  switchport port-security aging type inactivity
  switchport port-security
  ip dhcp snooping limit rate 15
  load-interval 30
!
Template Name : PRINTER_INTERFACE_TEMPLATE
Modified : No
Template Definition :
  spanning-tree portfast
  spanning-tree bpduguard enable
  switchport mode access
  switchport port-security maximum 2
  load-interval 60
!
Template Name : ROUTER_INTERFACE_TEMPLATE
Modified : No
Template Definition :
  spanning-tree portfast trunk
  spanning-tree bpduguard enable
  switchport mode trunk
  mls qos trust dscp
  srr-queue bandwidth share 1 30 35 5
  priority-queue out
!
Template Name : SWITCH_INTERFACE_TEMPLATE
Modified : No
Template Definition :
  switchport mode trunk
mls qos trust cos
srr-queue bandwidth share 1 30 35 5
priority-queue out

Template Name: TP_INTERFACE_TEMPLATE
Modified: No
Template Definition:
spanning-tree portfast
spanning-tree bpdu-guard enable
switchport mode access
switchport port-security maximum 3
switchport port-security maximum 2 vlan access
switchport port-security violation restrict
switchport port-security aging time 2
switchport port-security aging type inactivity
switchport port-security
storm-control broadcast level pps 1k
storm-control multicast level pps 2k
storm-control action trap
ip dhcp snooping limit rate 15
load-interval 30

End

Verifying all Interface Templates Binding for all templates

Device# show template interface binding all
<table>
<thead>
<tr>
<th>Template-name</th>
<th>Source</th>
<th>Method</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP_PHONE_INTERFACE_TEMPLATE</td>
<td>Built-in</td>
<td>Dynamic</td>
<td>Gi1/0/1, Gi1/0/2, Gi1/0/3, Gi1/0/4, Gi1/0/5, Gi1/0/6, Gi1/0/7, Gi1/0/8, Gi1/0/9, Gi1/0/10, Gi1/0/11, Gi1/0/12, Gi1/0/13, Gi1/0/14, Gi1/0/15, Gi1/0/16, Gi1/0/17, Gi1/0/18, Gi1/0/19, Gi1/0/20, Gi1/0/21, Gi1/0/22, Gi1/0/23, Gi1/0/24, Gi1/1/1, Gi1/1/2, Gi1/1/3</td>
</tr>
<tr>
<td>IP_PHONE_INTERFACE_TEMPLATE</td>
<td>Built-in</td>
<td>Static</td>
<td>Gi4/0/4</td>
</tr>
</tbody>
</table>

Verifying Static Template Binding for a Target Interface

Device# show template interface binding target GigabitEthernet 1/0/4
<table>
<thead>
<tr>
<th>Interface</th>
<th>Method</th>
<th>Source</th>
<th>Template</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gi1/0/4</td>
<td>Dynamic</td>
<td>built-in</td>
<td>IP_PHONE_INTERFACE_TEMPLATE</td>
</tr>
<tr>
<td></td>
<td>Static</td>
<td>user</td>
<td>TEST</td>
</tr>
<tr>
<td></td>
<td>Dynamic</td>
<td>Modified-built-in</td>
<td>TEST</td>
</tr>
</tbody>
</table>

Verifying Dynamic Template Binding for all templates

Device# show template service all

User-defined template:

Template Name: SVC-1
Template Definition:
vlan 100
access-group acl1

Built-in template:

Template Name: SVC-2
Template Definition:
vlan 100
access-group acl1
aaa downloaded template:
Template Name : SVC-2
Template Definition:
vlan 100
default access
access-group acl1

Verifying Template Binding for a Target Interface
Device# show template binding target GigabitEthernet 1/0/4

<table>
<thead>
<tr>
<th>Interface Templates:</th>
<th>method</th>
<th>Source</th>
<th>Template</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gi1/0/4</td>
<td>Dynamic</td>
<td>built-in</td>
<td>IP_PHONE_INTERFACE_TEMPLATE</td>
</tr>
<tr>
<td></td>
<td>Static</td>
<td>user</td>
<td>TEST</td>
</tr>
<tr>
<td></td>
<td>Dynamic</td>
<td>Modified-built-in</td>
<td>TEST</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Service Templates:</th>
<th>Source</th>
<th>Session-Mac</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVC1</td>
<td>user</td>
<td>aa-bb-cc-dd-ee-ff</td>
</tr>
<tr>
<td>SVC2</td>
<td>built-in</td>
<td>ab-ab-ab-ab-ab-ab</td>
</tr>
<tr>
<td>SVC3</td>
<td>aaa</td>
<td>ac-ac-ac-ac-ac-ac</td>
</tr>
</tbody>
</table>

Configuration Examples for Interface Templates

Example: Configuring User Interface Templates

Example: Configuring User Templates
Device# enable
Device (config)# configure terminal
Device(config)# template user-template1
Device(config-template)# load-interval 60
Device(config-template)# description This is a user template
Device(config-template)# keepalive 60
Device(config)# end

Example: Sourcing Interface Templates
Device> enable
Device# configure terminal
Device(config)# interface fastethernet 4/0/0
Device(config-if)# source template user-template1
Device(config-if)# end

Example: Dynamically Binding Interface Templates
Device> enable
Device# configure terminal
Device(config)# interface GigabitEthernet 4/0/1
Device(config-if)# service-policy type control subscriber POLICY_Gi1/0/12
Device(config-if)# end
Additional References for Interface Templates

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS commands</td>
<td>Cisco IOS Master Command List, All Releases</td>
</tr>
<tr>
<td>Identity-Based Networking Services commands</td>
<td>Cisco IOS Identity-Based Networking Services Command Reference</td>
</tr>
<tr>
<td>Autoconf</td>
<td>&quot;Autoconf&quot; module in Identity-Based Networking Services Configuration Guide.</td>
</tr>
</tbody>
</table>

Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.</td>
<td><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></td>
</tr>
</tbody>
</table>

Feature Information for Interface Templates

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.
### Table 7: Feature Information for Interface Templates

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interface Templates</td>
<td>Cisco IOS 15.2(1)SY</td>
<td>An interface template provides a mechanism to configure multiple commands at the same time and associate it with a target such as an interface. In Cisco IOS 15.2(1)SY, this feature was supported on Cisco Catalyst 6500 Series Switches. The following commands were introduced or modified: access-session, authentication, carrier-delay, dampening, default, description, dot1x, hold-queue, ip, keepalive, load-interval, mab, mls, peer, priority-queue, queue-set, radius-server, service-policy type control subscriber, source, spanning-free, storm-control, subscriber, switchport, trust.</td>
</tr>
</tbody>
</table>
Autoconf

Autoconf is a solution that can be used to manage port configurations for data or voice VLAN, quality of service (QoS) parameters, storm control, and MAC-based port security on end devices that are deployed in the access layer of a network.

• Finding Feature Information, page 81
• Restrictions for Autoconf, page 81
• Information About Autoconf, page 82
• How to Configure Autoconf, page 87
• Configuration Examples for Autoconf, page 95
• Additional References for Autoconf, page 95
• Feature Information for Autoconf, page 96

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions for Autoconf

• Interface templates are not applicable for wireless sessions.

• When the Autoconf feature is enabled using the autoconf enable command, the default Autoconf service policy is applied to all interfaces. No other service policy can be applied globally using the service-policy command. To apply a different service policy, you must disable Autoconf on that interface. When a service policy is applied globally, you must disable it before enabling the Autoconf feature.
• When both local (interface-level) and global service policies exist, the local policy take precedence. Events in the local service policy are handled and the global service policy is not applied. The global service policy comes into effect only when the local policy is removed.

• Service templates cannot be applied to interfaces, and interface templates cannot be applied to service instances.

• Only one service template can be nested inside an interface template.

### Information About Autoconf

#### Benefits of Autoconf

The Autoconf feature permits hard binding between the end device and the interface. Autoconf falls under the umbrella of the Smart Operations solution. Smart Operations is a comprehensive set of capabilities that can simplify and improve LAN switch deployment. Smart Operations help organizations deliver operational excellence and scale services on the network.

The Autoconf feature automatically applies the needed configurations on the device ports to enable the efficient performance of each directly connected end device using a set of interface configurations that are configured inside an interface template.

- Autoconf efficiently applies commands to an interface because the parser does not need to parse each command each time.

- Configurations that are applied through the Autoconf feature can be reliably removed from a port without impacting previous or subsequent configurations on the port.

- The Autoconf feature provides built-in and user-defined configurations using interface and service templates. Configurations applied through templates can be centrally updated with a single operation.

- Using the Autoconf feature, a configuration can be applied to ports and access sessions.

- The Autoconf feature reduces ongoing maintenance for devices and attached end devices by making them intuitive and autoconfigurable. This reduces operation expenses (OPEX) and lowers the total cost of ownership (TCO).

#### Identity Session Management and Templates

A key advantage of the Autoconf feature is that the core session management capability is decoupled from the application-specific logic; thus, allowing the same framework to be used regardless of the criteria for policy determination or the nature of the policies applied.

The identity session management infrastructure allows configurations and/or policies to be applied as templates. Both service and interface templates are named containers of configuration and policy. Service templates may be applied only to access sessions, while interface templates may be applied only to ports. When a service template is applied to an access session, the contained configuration/policy is applied only to the target session and has no impact on other sessions that may be hosted on the same access port. Similarly, when an interface template is applied to an access port, it impacts all traffic exchanged on the port.
The Autoconf feature uses a set of built-in maps and built-in templates. The built-in templates are designed based on best practices for interface configurations. Built-in templates can be modified by the user to include customized configurations, limiting the need to create a new template.

The templates created by users are referred to as user-defined templates. User-defined templates can be defined on the device and can be mapped to any built-in or user-defined trigger.

Use the `show derived-config` command, to view the overall applied configurations applied by Autoconf template and manual configuration. The interface commands shown in the output of `show running-config interface type number` command are not necessarily the operational configuration. The Autoconf feature dynamically applies a template to the interface, and overrides any conflicting static configuration that is already applied.

## Autoconf Operation

Autoconf uses the Device Classifier to identify the end devices that are connected to a port.

The Autoconf feature uses the device classification information gleaned from Cisco Discovery Protocol, LLDP, DHCP, MAC addresses, and the Organizationally Unique Identifier (OUI) that is identified by the Device Classifier.

The Device Classifier provides improved device classification capabilities and accuracy, and increased device visibility for enhanced configuration management.

Device classification is enabled when you enable the Autoconf feature using `autoconf enable` command in global configuration mode.

The device detection acts as an event trigger, which in turn applies the appropriate automatic template to the interface.

The Autoconf feature is based on a three-tier hierarchy.

- A policy map identifies the trigger type for applying the Autoconf feature.
- A parameter map identifies the appropriate template that must be applied, based on the end device.
- The templates contain the configurations to be applied.

The Autoconf built-in templates and triggers perform these three steps automatically.

The Autoconf feature provides the following built-in templates:

- `AP_INTERFACE_TEMPLATE`
- `DMP_INTERFACE_TEMPLATE`
- `IP_CAMERA_INTERFACE_TEMPLATE`
- `IP_PHONE_INTERFACE_TEMPLATE`
- `LAP_INTERFACE_TEMPLATE`
- `MSP_CAMERA_INTERFACE_TEMPLATE`
- `MSP_VC_INTERFACE_TEMPLATE`
- `PRINTER_INTERFACE_TEMPLATE`
- `ROUTER_INTERFACE_TEMPLATE`
- `SWITCH_INTERFACE_TEMPLATE`
By default built-in templates are not displayed under running configuration. The built-in templates show in the running configuration only if you edit them.

The template that is selected is based on parameter map information applied to an interface. This information can be based on the following criteria:

- End Device type
- MAC address
- OUI
- User role
- Username

The Autoconf feature provides one built-in parameter map BUILTIN_DEVICE_TO_TEMPLATE with the following configuration:

Parameter-map name: BUILTIN_DEVICE_TO_TEMPLATE
Map: 10 map device-type regex "Cisco-IP-Phone"
  Action(s):
  20 interface-template IP_PHONE_INTERFACE_TEMPLATE
Map: 20 map device-type regex "Cisco-IP-Camera"
  Action(s):
  20 interface-template IP_CAMERA_INTERFACE_TEMPLATE
Map: 30 map device-type regex "Cisco-DMP"
  Action(s):
  20 interface-template DMP_INTERFACE_TEMPLATE
Map: 40 map oui eq "00.0f.44"
  Action(s):
  20 interface-template DMP_INTERFACE_TEMPLATE
Map: 50 map oui eq "00.23.ac"
  Action(s):
  20 interface-template DMP_INTERFACE_TEMPLATE
Map: 60 map device-type regex "Cisco-AIR-AP"
  Action(s):
  20 interface-template AP_INTERFACE_TEMPLATE
Map: 70 map device-type regex "Cisco-AIR-LAP"
  Action(s):
  20 interface-template LAP_INTERFACE_TEMPLATE
Map: 80 map device-type regex "Cisco-TelePresence"
  Action(s):
  20 interface-template TP_INTERFACE_TEMPLATE
Map: 90 map device-type regex "Surveillance-Camera"
  Action(s):
  10 interface-template MSP_CAMERA_INTERFACE_TEMPLATE
Map: 100 map device-type regex "Video-Conference"
  Action(s):
  10 interface-template MSP_VC_INTERFACE_TEMPLATE

Use the `show parameter-map type subscriber attribute-to-service All` command to view the configuration for the built-in parameter map.

The Autoconf feature provides one built-in policy map BUILTIN_AUTOCONF_POLICY with the following configuration:

BUILTIN_AUTOCONF_POLICY
  event identity-update match-all
10 class always do-until-failure
10 map attribute-to-service table BUILTIN_DEVICE_TO_TEMPLATE

Note

Use the **show policy-map type control subscriber BUILTIN_AUTOCONF_POLICY** command to view the configuration for the built-in policy map.

You can also manually create policy maps, parameter maps, and templates. When a trigger is created that is based on specific user information, a local 802.1X Cisco Identity Services Engine (ISE) server authenticates it ensuring the security of the operation.

An interface template can be dynamically activated (on an interface) using any of the following methods:

- **RADIUS CoA**—While Change of Authorization (CoA) commands are targeted to one or more access sessions, any referenced template must be applied to the interface hosting the referenced session.
- **RADIUS Access-Accept for client authentication or authorization**—Any referenced interface template returned in an Access-Accept must be applied to the port that is hosting the authorized access session.
- **Service template**—If an interface template is referenced in a service template that is either locally defined or sourced from the AAA server, the interface template must be applied to the interface hosting any access-session on which the service template is applied (add a new command for interface template reference from within a locally defined service template).
- **Subscriber control-policy action**—A mapping action under the subscriber control policy activates service and/or interface template (as referenced in a parameter map) based on the type of filter, and removes any templates associated with a previous policy.
- **Device-to-template parameter map**—A subscriber parameter map that allows the filter type to service and/or interface template mappings to be specified in an efficient and readable manner.

## Advantages of Using Templates

Using templates for autoconfiguration has the following benefits:

- Templates are parsed once when they are being defined. This makes dynamic application of the templates very efficient.
- Templates can be applied to an Ethernet interface that is connected to an end device, based on the type of the end device.
- Service templates allow the activation of session-oriented features, whereas interface templates apply configurations to the interface that is hosting a session.
- Service templates are applied to access sessions and hence only impact the traffic exchanged with a single endpoint on a port.
- Startup and running configurations of the device are not modified by the dynamic application of the template.
- Policy application is synchronized with the access-session life cycle, which is tracked by the framework by using all available techniques, including just link-up/link-down.
- Templates can be updated with a single operation. All applied instances of the templates are updated.
- Constituent commands of the templates do not appear in the running configuration.
• Templates can be removed with no impact on previous or subsequent configurations.

• Template application is acknowledged, allowing for synchronization and performing remedial actions where failures occur.

• Data VLAN, quality of service (QoS) parameters, storm control, and MAC-based port security are configured automatically based on the end device that is connected to the switch.

• The switch port is cleaned up completely by removing configurations when the device is disconnected from a port.

• Human error is reduced in the installation and configuration process.

Autoconf Functionality

The Autoconf feature is disabled by default in global configuration mode. When you enable the Autoconf feature in global configuration mode, it is enabled by default at the interface level. The built-in template configurations are applied based on the end devices detected on all interfaces.

Use the `access-session inherit disable autoconf` command to manually disable Autoconf at the interface level, even when Autoconf is enabled at the global level.

If you disable Autoconf at the global level, all interface-level configurations are disabled.

<table>
<thead>
<tr>
<th>Global</th>
<th>Interface Level</th>
<th>AutoConf Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disable</td>
<td>Disable</td>
<td>No automatic configurations are applied when an end device is connected.</td>
</tr>
<tr>
<td>Enable</td>
<td>Enabled by default</td>
<td>If Autoconf is enabled at the global level, it is enabled at the interface level by default. Built-in template configurations are applied based on the end devices that are detected on all interfaces.</td>
</tr>
<tr>
<td>Enable</td>
<td>Disable</td>
<td>Enabled at global level. Disabled at interface level. No automatic configurations are applied when an end device is connected to the interface on which Autoconf is disabled.</td>
</tr>
</tbody>
</table>

Autoconf allows you to retain the template even when the link to the end device is down or the end device is disconnected, by configuring the Autoconf sticky feature. Use the `access-session interface-template sticky` command to configure the Autoconf sticky feature in global configuration mode. The Autoconf sticky feature avoids the need for detecting the end device and applying the template every time the link flaps or device is removed and connected back.

The `access-session interface-template sticky` command is mandatory to apply an inbuilt template that contains `access-session` commands on an interface. Configure the `access-session interface-template sticky` command to apply interface template on a port using a service policy.

If you want to disable the Autoconf feature on a specific interface, use the `access-session inherit disable interface-template-sticky` command in interface configuration mode.
How to Configure Autoconf

Applying a Built-in Template to an End Device

The following task shows how to apply a built-in template on an interface that is connected to an end device, for example, a Cisco IP phone.

Before You Begin

Make sure that the end device, for example, a Cisco IP phone, is connected to the switch port.

SUMMARY STEPS

1. enable
2. configure terminal
3. autoconf enable
4. end
5. (Optional) show device classifier attached interface interface-type interface-number
6. show template binding target interface-type interface-number

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td>enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example: Device&gt; enable</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td>configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example: Device(config)# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td></td>
</tr>
<tr>
<td>autoconf enable</td>
<td>Enables the Autoconf feature.</td>
</tr>
<tr>
<td>Example: Device(config)# autoconf enable</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td></td>
</tr>
<tr>
<td>end</td>
<td>Exits global configuration mode and enters privileged EXEC mode.</td>
</tr>
<tr>
<td>Example: Device(config)# end</td>
<td></td>
</tr>
<tr>
<td>Step 5</td>
<td>Command or Action</td>
</tr>
<tr>
<td>--------</td>
<td>------------------</td>
</tr>
<tr>
<td></td>
<td>(Optional) show device classifier attached interface</td>
</tr>
<tr>
<td></td>
<td>interface-type interface-number</td>
</tr>
<tr>
<td>Example:</td>
<td>Device# show device classifier attached interface Gi3/0/26</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 6</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>show template binding target interface-type interface-number</td>
<td>Displays the configuration applied through the template on the interface.</td>
</tr>
<tr>
<td>Example:</td>
<td>Device# show template binding target gi3/0/26</td>
<td></td>
</tr>
</tbody>
</table>

**Verifying the device classification of an End Device**

The following example shows that an IP phone is classified by the Device Classifier with correct attributes:

Device# show device classifier attached interface GigabitEthernet 3/0/26

Summary:

<table>
<thead>
<tr>
<th>MAC_Address</th>
<th>Port_Id</th>
<th>Profile Name</th>
<th>Device Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>0026.0bd9.7bbb</td>
<td>Gi3/0/26</td>
<td>Cisco-IP-Phone-7962</td>
<td>Cisco IP Phone 7962</td>
</tr>
</tbody>
</table>

**Verifying the Interface Template on an Interface**

The following example shows that a built-in interface template is applied on the interface:

Device# show template binding target GigabitEthernet 3/0/26

Interface Templates

<table>
<thead>
<tr>
<th>Interface: Gi4/0/11</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
</tr>
<tr>
<td>--------</td>
</tr>
<tr>
<td>dynamic</td>
</tr>
</tbody>
</table>

**Verifying the Interface Configuration**

The following example shows how to verify the interface configuration after the interface template is applied to the IP phone connected to the GigabitEthernet interface 3/0/26:

Device# show running-config interface GigabitEthernet 3/0/26

Building configuration...

Current configuration : 624 bytes

interface GigabitEthernet3/0/26

End

Device# show derived-config interface GigabitEthernet 3/0/26

Building configuration...

Derived configuration : 649 bytes
interface GigabitEthernet3/0/26
  switchport mode access
  switchport block unicast
  switchport port-security maximum 3
  switchport port-security maximum 2 vlan access
  switchport port-security violation restrict
  switchport port-security aging time 2
  switchport port-security aging type inactivity
  switchport port-security
  load-interval 30
  storm-control broadcast level pps 1k
  storm-control multicast level pps 2k
  storm-control action trap
  spanning-tree portfast
  spanning-tree bpduguard enable
  service-policy input AutoConf=4.0-CiscoPhone-Input-Policy
  service-policy output AutoConf=4.0-Output-Policy
  ip dhcp snooping limit rate 15
end

Verifying Interface Configuration for Cisco IOS 4500 Series, Cisco IOS 3650 Series, Cisco IOS 3560 Series, and Cisco IOS 2960 Switches

The following example shows how to verify the interface configuration:

Device# show template interface source built-in all
Building configuration...

Template Name : AP_INTERFACE_TEMPLATE
Modified : No
Template Definition :
  switchport mode trunk
  switchport nonegotiate

!
Template Name : DMP_INTERFACE_TEMPLATE
Modified : No
Template Definition :
  spanning-tree portfast
  spanning-tree bpduguard enable
  switchport mode access
  switchport block unicast
  switchport port-security

!
Template Name : IP_CAMERA_INTERFACE_TEMPLATE
Modified : No
Template Definition :
  spanning-tree portfast
  spanning-tree bpduguard enable
  switchport mode access
  switchport block unicast
  switchport port-security

!
Template Name : IP_PHONE_INTERFACE_TEMPLATE
Modified : No
Template Definition :
  spanning-tree portfast
  spanning-tree bpduguard enable
  switchport mode access
  switchport block unicast
  switchport port-security maximum 3
  switchport port-security maximum 2 vlan access
  switchport port-security violation restrict
  switchport port-security aging time 2
  switchport port-security aging type inactivity
  switchport port-security
  load-interval 30
  storm-control broadcast level pps 1k
  storm-control multicast level pps 2k
  storm-control action trap
Applying a Built-in Template to an End Device

<table>
<thead>
<tr>
<th>Template Name</th>
<th>Template Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>LAP_INTERFACE_TEMPLATE</td>
<td>spanning-tree portfast</td>
</tr>
<tr>
<td></td>
<td>spanning-tree bpdu guard enable</td>
</tr>
<tr>
<td></td>
<td>switchport mode access</td>
</tr>
<tr>
<td></td>
<td>switchport block unicast</td>
</tr>
<tr>
<td></td>
<td>switchport port-security violation protect</td>
</tr>
<tr>
<td></td>
<td>switchport port-security aging time 2</td>
</tr>
<tr>
<td></td>
<td>switchport port-security aging type inactivity</td>
</tr>
<tr>
<td></td>
<td>switchport port-security violation restrict</td>
</tr>
<tr>
<td></td>
<td>storm-control broadcast level pps 1k</td>
</tr>
<tr>
<td></td>
<td>storm-control multicast level pps 2k</td>
</tr>
<tr>
<td></td>
<td>storm-control action trap</td>
</tr>
</tbody>
</table>

| MSP_CAMERA_INTERFACE_TEMPLATE | spanning-tree portfast                                   |
|                                | spanning-tree bpdu guard enable                           |
|                                | switchport mode access                                    |
|                                | switchport block unicast                                  |
|                                | switchport port-security                                  |

| MSP_VC_INTERFACE_TEMPLATE     | spanning-tree portfast                                   |
|                                | spanning-tree bpdu guard enable                           |
|                                | switchport mode access                                    |
|                                | switchport block unicast                                  |
|                                | switchport port-security violation restrict               |
|                                | switchport port-security aging time 2                    |
|                                | switchport port-security aging type inactivity           |
|                                | switchport port-security                                  |
|                                | ip dhcp snooping limit rate 15                            |
|                                | load-interval 30                                          |

| PRINTER_INTERFACE_TEMPLATE    | spanning-tree portfast                                   |
|                                | spanning-tree bpdu guard enable                           |
|                                | switchport mode access                                    |
|                                | switchport port-security maximum 2                       |
|                                | switchport port-security load-interval 60                |

| ROUTER_INTERFACE_TEMPLATE     | spanning-tree portfast trunk                             |
|                                | spanning-tree bpdu guard enable                           |
|                                | switchport mode trunk                                     |

| SWITCH_INTERFACE_TEMPLATE     | switchport mode trunk                                     |

| TP_INTERFACE_TEMPLATE         | spanning-tree portfast                                   |
|                                | spanning-tree bpdu guard enable                           |
|                                | switchport mode access                                    |
|                                | switchport port-security maximum 3                       |
|                                | switchport port-security maximum 2 vlan access            |
|                                | switchport port-security violation restrict               |
switchport port-security aging time 2
switchport port-security aging type inactivity
switchport port-security
storm-control broadcast level pps 1k
storm-control multicast level pps 2k
storm-control action trap
ip dhcp snooping limit rate 15
load-interval 30
!

Verifying Global Configuration after Applying Autoconf

Device# show running config
class-map match-any AutoConf-4.0-Scavenger-Queue
  match dscp cs1
  match cos 1
  match access-group name AutoConf-4.0-ACL-Scavenger
class-map match-any AutoConf-4.0-VoIP
  match dscp ef
  match cos 5
class-map match-any AutoConf-4.0-Control-Mgmt-Queue
  match cos 3
  match dscp cs7
  match dscp cs6
  match dscp cs3
  match dscp cs2
  match access-group name AutoConf-4.0-ACL-Signaling
class-map match-any AutoConf-4.0-Multimedia-Conf
  match dscp af41
  match dscp af42
  match dscp af43
class-map match-all AutoConf-4.0-Broadcast-Vid
  match dscp cs5
class-map match-any AutoConf-4.0-Bulk-Data
  match dscp af11
  match dscp af12
  match dscp af13
class-map match-all AutoConf-4.0-Realtime-Interact
  match dscp cs4
class-map match-any AutoConf-4.0-VoIP-Signal
  match dscp cs3
  match cos 3
class-map match-any AutoConf-4.0-Trans-Data-Queue
  match cos 2
  match dscp af21
  match dscp af22
  match dscp af23
  match access-group name AutoConf-4.0-ACL-Transactional-Data
class-map match-any AutoConf-4.0-VoIP-Data
  match dscp ef
  match cos 5
class-map match-any AutoConf-4.0-Multimedia-Stream
  match dscp af31
  match dscp af32
  match dscp af33
class-map match-all AutoConf-4.0-Internetwork-Ctrl
  match dscp cs6
class-map match-all AutoConf-4.0-VoIP-Signal-Cos
  match cos 3
class-map match-any AutoConf-4.0-Multimedia-Stream-Queue
  match dscp af31
  match dscp af32
  match dscp af33
class-map match-all AutoConf-4.0-Network-Mgmt
  match dscp cs2
class-map match-all AutoConf-4.0-VoIP-Data-Cos
  match cos 5
class-map match-any AutoConf-4.0-Priority-Queue
  match cos 5
  match dscp ef
  match dscp cs5
  match dscp cs4
class-map match-any AutoConf-4.0-Bulk-Data-Queue
  match cos 1
  match dscp af11
  match dscp af12
  match dscp af13
  match access-group name AutoConf-4.0-ACL-Bulk-Data
class-map match-any AutoConf-4.0-Transaction-Data
  match dscp af21
  match dscp af22
  match dscp af23
class-map match-any AutoConf-4.0-Multimedia-Conf-Queue
  match cos 4
  match dscp af41
  match dscp af42
  match dscp af43
  match access-group name AutoConf-4.0-ACL-Multimedia-Conf
class-map match-all AutoConf-4.0-Network-Ctrl
  match dscp cs7
class-map match-all AutoConf-4.0-Scavenger
  match dscp cs1
class-map match-any AutoConf-4.0-Signaling
  match dscp cs3
  match cos 3
!
!
policy-map AutoConf-4.0-Cisco-Phone-Input-Policy
class AutoConf-4.0-VoIP-Data-Cos
set dscp ef
  police cir 128000 bc 8000
  exceed-action set-dscp-transmit cs1
  exceed-action set-cos-transmit 1
class AutoConf-4.0-VoIP-Signal-Cos
set dscp cs3
  police cir 32000 bc 8000
  exceed-action set-dscp-transmit cs1
  exceed-action set-cos-transmit 1
class class-default
  set dscp default
  set cos 0
!
policy-map AutoConf-4.0-Output-Policy
class AutoConf-4.0-Scavenger-Queue
  bandwidth remaining percent 1
class AutoConf-4.0-Priority-Queue
  priority
  police cir percent 30 bc 33 ms
class AutoConf-4.0-Control-Mgmt-Queue
  bandwidth remaining percent 10
class AutoConf-4.0-Multimedia-Conf-Queue
  bandwidth remaining percent 10
class AutoConf-4.0-Multimedia-Stream-Queue
  bandwidth remaining percent 10
class AutoConf-4.0-Trans-Data-Queue
  bandwidth remaining percent 10
dbl
class AutoConf-4.0-Bulk-Data-Queue
  bandwidth remaining percent 4
dbl
class class-default
  bandwidth remaining percent 25
dbl
!
policy-map AutoConf-DMP
class class-default
  set dscp cs2
policy-map AutoConf-IPVSC
class class-default
  set cos dscp table AutoConf-DscpToCos
class AutoConf-4.0-VoIP
class AutoConf-4.0-Broadcast-Vid
class AutoConf-4.0-Realtime-Interact
class AutoConf-4.0-Network-Ctrl
class AutoConf-4.0-Internetwork-Ctrl
class AutoConf-4.0-Signaling
Applying a Modified Built-in Template to an End Device

The following task shows how to modify a built-in template when multiple wireless access points and IP cameras are connected to a switch.

**SUMMARY STEPS**

1. **enable**
2. **configure terminal**
3. **template** *template-name*
4. **switchport access vlan** *vlan-id*
5. **description** *description*
6. **exit**
7. **autoconf enable**
8. **end**
9. **show template interface binding all**

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example: Device&gt; enable</td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example: Device(config)# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> template <em>template-name</em></td>
<td>Enters template configuration mode for the builtin template.</td>
</tr>
<tr>
<td>Example: Device(config)# template AP_INTERFACE_TEMPLATE</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> switchport access vlan <em>vlan-id</em></td>
<td>Sets the VLAN when the interface is in access mode.</td>
</tr>
<tr>
<td>Example: Device(config-template)# switchport access vlan 20</td>
<td></td>
</tr>
</tbody>
</table>
### Purpose

#### Command or Action

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>Example</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 5</td>
<td>description</td>
<td><code>description</code> modified AP</td>
<td>Modifies the description of the built-in template.</td>
</tr>
<tr>
<td>Example:</td>
<td>Device(config-template)# description modified AP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Step 6</td>
<td>exit</td>
<td><code>exit</code></td>
<td>Exits template configuration mode and enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>Device(config-template)# exit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Step 7</td>
<td>autoconf enable</td>
<td><code>autoconf enable</code></td>
<td>Enables the Autoconf feature.</td>
</tr>
<tr>
<td>Example:</td>
<td>Device(config)# autoconf enable</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Step 8</td>
<td>end</td>
<td><code>end</code></td>
<td>Exits global configuration mode and enters privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>Device(config)# end</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Step 9</td>
<td>show template interface binding all</td>
<td><code>show template interface binding all</code></td>
<td>Displays whether the template is applied on the interface.</td>
</tr>
<tr>
<td>Example:</td>
<td>Device# show template interface binding all</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Verifying the Device classification of an End Device

The following example shows that the IP camera and access points are classified by the Device Classifier with correct attributes:

Device# show device classifier attached detail

DC default profile file version supported = 1

<table>
<thead>
<tr>
<th>MAC_Address</th>
<th>Port_Id</th>
<th>Cert</th>
<th>Parent</th>
<th>Proto</th>
<th>ProfileType</th>
<th>Profile Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>001d.a1ef.23a8</td>
<td>Gi1/0/7</td>
<td>30</td>
<td>C</td>
<td>M</td>
<td>Default</td>
<td>Cisco-AIR-AP-1130</td>
</tr>
<tr>
<td>AIR-AP1131AG-A-K9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>001e.7a26.eb05</td>
<td>Gi1/0/30</td>
<td>70</td>
<td>C</td>
<td>M</td>
<td>Default</td>
<td>Cisco-IP-Camera</td>
</tr>
<tr>
<td>IP Camera</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Verifying the Interface Template on an Interface

The following example shows that a built-in interface template is applied on the interface:

Device# show template interface binding all

<table>
<thead>
<tr>
<th>Template-Name</th>
<th>Source</th>
<th>Method</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP_CAMERA_INTERFACE_TEMPLATE</td>
<td>Built-in</td>
<td>dynamic</td>
<td>Gi1/0/30</td>
</tr>
<tr>
<td>AP_INTERFACE_TEMPLATE</td>
<td>Modified-Built-in</td>
<td>dynamic</td>
<td>Gi1/0/7</td>
</tr>
</tbody>
</table>
Configuration Examples for Autoconf

Example: Applying a Built-in Template to an End Device

The following example shows how to apply a built-in template to an end device connected to an interface.

Device> enable
Device(config)# configure terminal
Device(config)# autoconf enable
Device(config)# end
Device# show device classifier attached interface Gi3/0/26
Device# show template binding target GigabitEthernet 3/0/26

Example: Applying a Modified Built-in Template to an End Device

The following example shows how to modified built-in template and verify the configuration:

Device> enable
Device(config)# configure terminal
Device(config)# template AP_INTERFACE_TEMPLATE
Device(config-template)# switchport access vlan 20
Device(config-template)# description modifiedAP
Device(config-template)# exit
Device(config)# autoconf enable
Device(config)# end
Device# show template interface binding all

Additional References for Autoconf

Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS commands</td>
<td>Cisco IOS Master Command List, All Releases</td>
</tr>
<tr>
<td>Cisco identity-based networking services commands</td>
<td>Cisco IOS Identity-Based Networking Services Command Reference</td>
</tr>
<tr>
<td>Interface Templates</td>
<td>“Interface Templates” module in Identity-Based Networking Services Configuration Guide</td>
</tr>
</tbody>
</table>

Standards and RFCs

<table>
<thead>
<tr>
<th>Standard/RFC</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEEE 802.1X</td>
<td>Port Based Network Access Control</td>
</tr>
</tbody>
</table>
Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support website provides extensive online resources, including</td>
<td><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></td>
</tr>
<tr>
<td>documentation and tools for troubleshooting and resolving technical issues</td>
<td></td>
</tr>
<tr>
<td>with Cisco products and technologies.</td>
<td></td>
</tr>
<tr>
<td>To receive security and technical information about your products, you</td>
<td></td>
</tr>
<tr>
<td>can subscribe to various services, such as the Product Alert Tool (accessed</td>
<td></td>
</tr>
<tr>
<td>from Field Notices), the Cisco Technical Services Newsletter, and Really</td>
<td></td>
</tr>
<tr>
<td>Simple Syndication (RSS) Feeds.</td>
<td></td>
</tr>
<tr>
<td>Access to most tools on the Cisco Support website requires a Cisco.com user</td>
<td></td>
</tr>
<tr>
<td>ID and password.</td>
<td></td>
</tr>
</tbody>
</table>

Feature Information for Autoconf

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.
Table 8: Feature Information for Autoconf

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Autoconf</td>
<td>Cisco IOS 15.2(1)SY</td>
<td>Autoconf is a solution that can be used to manage port configurations for data or voice VLANs, quality of QoS parameters, storm control, and MAC-based port security on end devices that are deployed in the access layer of a network. The Autoconf feature automatically applies the configurations needed on the device ports to enable the efficient performance of each directly connected end device using a set of interface configurations that are configured inside an interface template. This mechanism ensures that no configurations are needed from the end device. In Cisco IOS 15.2(1)SY, this feature was supported on Cisco Catalyst 6500 Series Switches. The following commands were added or modified: autoconf enable, map attribute-to-service (autoconf), map device-type (service-template), parameter-map type subscriber (service-template), show parameter-map type subscriber attribute-to-service all, show template interface.</td>
</tr>
</tbody>
</table>