CAC for IPv6 Flows

Last Updated: January 15, 2013

The CAC for IPv6 Flows feature provides IPv6 support for Resource Reservation Protocol (RSVP). By enabling this feature, the network is made to support the complete RSVP IPv6 functionality for Call Admission Control (CAC) and Medianet.

- Finding Feature Information, page 1
- Prerequisites for CAC for IPv6 Flows, page 1
- Restrictions for CAC for IPv6 Flows, page 1
- Information About CAC for IPv6 Flows, page 2
- How to Configure CAC for IPv6 Flows, page 3
- Configuration Examples for CAC for IPv6 Flows, page 11
- Additional References, page 13
- Feature Information for CAC for IPv6 Flows, page 14

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 CAC for IPv6 Flows

You must configure RSVP on one or more interfaces on at least two neighboring routers that share a link within the network.

Restrictions for CAC for IPv6 Flows

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

- The RSVP functionality is not High Availability (HA) compliant; hence, the CAC for IPv6 Flows feature is noncompliant with HA.
- Multiprotocol Label Switching (MPLS) virtual private network (VPN) Virtual Routing and Forwarding (VRF) instances are not supported.

Information About CAC for IPv6 Flows

The CAC for IPv6 Flows feature provides IPv6 support for RSVP, which allows, services that run RSVP as a transport protocol such as CAC, TE, mediatrace, and medianet, to be IPv6 compliant.

RSVP signaling can be initiated and terminated by the following entities:

- RSVP at the endpoint
- RSVP source or receiver proxy
- RSVP agent or application server
- RSVP proxy from the network device (router or switch)

To enable the CAC for IPv6 Flows feature, the endpoints and application servers are designed to be IPv6 systems that signal RSVP to the network.

- Differences Between IPv4 and IPv6 Flows, page 2
- IPv6 Support for RSVP Features, page 2

Differences Between IPv4 and IPv6 Flows

Following are the differences between IPv4 and IPv6 flows in an RSVP network:

- For general routing purposes, global IPv6 addresses are not required on all intermediate devices. Link local adddresses are used instead. However, global IPv6 addresses are required on ingress and egress interfaces.
- Link local addresses are used for neighbor authentication in an IPv6 network as opposed to how global IP addresses are used in an IPv4 network.

IPv6 Support for RSVP Features

The CAC for IPv6 Flows feature extends IPv6 support to the following RSVP features:

- CAC
- Transport Protocol
- RSVP policy support for global and interface configuration modes, except access control list (ACL) support
- RSVP authentication, except ACL support
- Previous hop (PHOP) overwrite in interface configuration mode
- Fast Local Repair (FLR)
- Ingress CAC
- Flexible bandwidth
- Virtual Routing and Forwarding (VRF)
- RSVP reliable messaging
- Flow Metadata

For more information about each of these features, see *QoS: RSVP Configuration Guide*.

How to Configure CAC for IPv6 Flows

- Adding Senders or Receivers for IPv6 Flows to the RSVP Database, page 3
- Configuring a Static Sender for IPv6 Flows, page 4
- Configuring a Static Receiver for IPv6 Flows, page 5
- Configuring a Receiver Proxy for IPv6 Flows on a Tailend Device, page 7
- Configuring RSVP as a Transport Protocol for IPv6 Flows, page 8
- Binding a Key Chain to an RSVP IPv6 Neighbor, page 9
- Configuring PHOP for IPv6 Flows, page 10

Adding Senders or Receivers for IPv6 Flows to the RSVP Database

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 rsvp sender** *session-ipv6-address sender-ipv6-address* [**tcp** | **udp** | *ip-protocol*] *session-dest-port sender-source-port previous-hop-ipv6-address previous-hop-interface bandwidth burst-size*
4. **ipv6 rsvp reservation** *session-ipv6-address sender-ipv6-address* [**tcp** | **udp** | *ip-protocol*] *session-dest-port sender-source-port next-hop-ipv6-address next-hop-interface* {**ff** | **se** | **wf**} {**rate** | **load**} *bandwidth burst-size*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**ipv6 rsvp sender** *session-ipv6-address sender-ipv6-address* [**tcp** | **udp** | *ip-protocol*] *session-dest-port sender-source-port previous-hop-ipv6-address previous-hop-interface bandwidth burst-size* **Example:** `Device(config)# ipv6 rsvp sender 2001:DB8:1:: 2001:DB8:2:: tcp 2 3 2001:DB8::1 fastEthernet 0/1 2 3`	Adds senders to the RSVP database. • Enables a networking device to behave like it is receiving and processing IPv6 RSVP PATH messages from the sender or previous hop routes containing the indicated attributes. • The related **ipv6 rsvp sender-host** command enables a device to simulate a host generating IPv6 RSVP PATH messages. It is used mostly for debugging and testing purposes.
Step 4	**ipv6 rsvp reservation** *session-ipv6-address sender-ipv6-address* [**tcp** | **udp** | *ip-protocol*] *session-dest-port sender-source-port next-hop-ipv6-address next-hop-interface* {**ff** | **se** | **wf**} {**rate** | **load**} *bandwidth burst-size* **Example:** `Device(config)# ipv6 rsvp reservation 2001:DB8:1:: 2001:DB8::2 tcp 2 3 2001:DB8::3 fastEthernet 0/1 ff load 2 4`	Adds receivers to the RSVP database and enables a device to behave like it is receiving and processing IPv6 RSVP RESV messages. • The related **ipv6 rsvp reservation-host** command enables a device to simulate a host generating RSVP RESV messages. It is used mostly for debugging and testing purposes.
Step 5	**end** **Example:** `Device(config)# end`	Exits global configuration mode and returns to privileged EXEC mode.

Configuring a Static Sender for IPv6 Flows

Perform this task to configure a static sender for IPv6 flows with a VRF on a headend device, to make the device proxy an IPv6 RSVP PATH message.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 rsvp sender-host** *session-ipv6-address sender-ipv6-address* {**tcp** | **udp** | *ip-protocol*} *session-destination-port sender-source-port bandwidth burst-size* [**identity** *alias*] [**vrf** *vrf-name*]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ipv6 rsvp sender-host** *session-ipv6-address sender-ipv6-address* {**tcp** | **udp** | *ip-protocol*} *session-destination-port sender-source-port bandwidth burst-size* [**identity** *alias*] [**vrf** *vrf-name*] **Example:** `Device(config)# ipv6 rsvp sender-host 2001:DB8:1:: 2001:DB8:2:: udp 1 1 10 10 vrf myvrf`	Enables a networking device to simulate a host generating IPv6 RSVP PATH messages. • The optional **identity** *alias*identity alias keyword and argument pair specifies an application ID alias. The string can have as many as 64 printable characters (in the range 0x20 to 0x7E). **Note** If you use the quotation marks (" ") or a question mark (?) as part of the alias string, you must type the CTRL/V key sequence before entering the embedded " " or ? characters. The alias is never transmitted to other networking devices.
Step 4	**end** **Example:** `Device(config)# end`	Exits global configuration mode and returns to privileged EXEC mode.

Configuring a Static Receiver for IPv6 Flows

Perform this task to configure a static RSVP receiver with an application ID to make the device proxy an IPv6 RSVP RESV message containing an application ID on behalf of an RSVP-unaware receiver application.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. Do one of the following:
   - **ipv6 rsvp reservation-host** *session-ipv6-address sender-ipv6-address* {**tcp** | **udp** | *ip-protocol*} *session-dest-port sender-source-port* {**ff** | **se** | **wf**} {**rate** | **load**} *bandwidth burst-size* [**identity** *alias*]
   - **ipv6 rsvp reservation** *session-ipv6-address sender-ipv6-address* {**tcp** | **udp** | *ip-protocol*} *session-dest-port sender-source-port next-hop-ipv6-address next-hop-interface* {**ff** | **se** | **wf**} {**rate** | **load**} *bandwidth burst-size* [**identity** *alias*]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

Command or Action	Purpose

Step 3 Do one of the following:
• **ipv6 rsvp reservation-host** *session-ipv6-address sender-ipv6-address* {**tcp** | **udp** | *ip-protocol*} *session-dest-port sender-source-port* {**ff** | **se** | **wf**} {**rate** | **load**} *bandwidth burst-size* [**identity** *alias*]
• **ipv6 rsvp reservation** *session-ipv6-address sender-ipv6-address* {**tcp** | **udp** | *ip-protocol*} *session-dest-port sender-source-port next-hop-ipv6-address next-hop-interface* {**ff** | **se** | **wf**} {**rate** | **load**} *bandwidth burst-size* [**identity** *alias*]
Example:
`Device(config)# ipv6 rsvp reservation-host 2001:DB8::1 2001:DB8::2 udp 20 30 se load 100 60 identity rsvp-voice`
Example:
`Device(config)# ipv6 rsvp reservation 2001:DB8:1:: FFFF:FFFF:: udp 20 0 172.16.4.1 Ethernet1 wf rate 350 65 identity xyz`
	Enables a device to simulate a host generating IPv6 RSVP RESV messages.
	• The optional **identity** *alias* keyword and argument pair specifies an application ID alias. The string can have as many as 64 printable characters (in the range 0x20 to 0x7E).
	Note If you use the quotation marks (" ") or a question mark (?) as part of the alias string, you must type the CTRL/V key sequence before entering the embedded " " or ? character. The alias is never transmitted to other devices.
	Note Use the **ipv6 rsvp reservation-host** command if the device is the destination or the **ipv6 rsvp reservation** command to have the device proxy on behalf of a downstream host.

Step 4 **end** **Example:** `Device(config)# end`	Exits global configuration mode and returns to privileged EXEC mode.

Configuring a Receiver Proxy for IPv6 Flows on a Tailend Device

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 rsvp listener** [**vrf** *vrf-name*] *dst* {**udp** | **tcp** | **any** | *number*} {**any** | *dst-port*} {**announce** | **reply** | **reject**}
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ipv6 rsvp listener** [**vrf** *vrf-name*] *dst* {**udp** | **tcp** | **any** | *number*} {**any** | *dst-port*} {**announce** | **reply** | **reject**} **Example:** `Device(config)# ipv6 rsvp listener vrf myvrf 2001:DB8:1:: any any reply`	Configures an RSVP device to listen for IPv6 PATH messages.
Step 4	**end** **Example:** `Device(config)# end`	(Optional) Returns to privileged EXEC mode.

Configuring RSVP as a Transport Protocol for IPv6 Flows

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 rsvp transport client** *client-id*
4. **ipv6 rsvp transport sender-host** [**tcp** | **udp**] *destination-ipv6-address source-ipv6-address ip-protocol dest-port source-port client-id init-id instance-id* [**vrf** *vrf-name*] [**data** *data-value*]
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ipv6 rsvp transport client** *client-id* **Example:** `Device(config)# ipv6 rsvp transport client 2`	Registers an RSVP transport client ID with RSVP for IPv6 flows. • This command is used for debugging and testing.
Step 4	**ipv6 rsvp transport sender-host** [**tcp** | **udp**] *destination-ipv6-address source-ipv6-address ip-protocol dest-port source-port client-id init-id instance-id* [**vrf** *vrf-name*] [**data** *data-value*] **Example:** `Device(config)# ipv6 rsvp transport sender-host tcp 2001:DB8:10:: 2001:DB:11:: 3 4 5 2 3 4 vrf vr1`	Creates an RSVP transport session, which enables a networking device to simulate a host generating IPv6 RSVP PATH message. • This command is used for debugging and testing purposes.
Step 5	**end** **Example:** `Device(config)# end`	Exits global configuration mode and returns to privileged EXEC mode.

Binding a Key Chain to an RSVP IPv6 Neighbor

Perform this task to bind a key chain to an RSVP IPv6 neighbor for neighbor authentication.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 rsvp authentication neighbor address** *ipv6-address* **key-chain** *key-chain-name*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ipv6 rsvp authentication neighbor address** *ipv6-address* **key-chain** *key-chain-name* **Example:** `Device(config)# ipv6 rsvp authentication neighbor address 2001:db8:1:: 1 key-chain neighbor_V`	Binds a key chain to an IPv6 address or to an ACL and enters key-chain mode.
Step 4	**end** **Example:** `Device(config-keychain)# end`	Returns to privileged EXEC mode.

Configuring PHOP for IPv6 Flows

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip rsvp bandwidth** [*interface-kbps*] [*single-flow-kbps*]
5. **ipv6 rsvp source** {**address** *ipv6-address* | **interface** *type number*}
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Device(config)# interface Ethernet0/0`	Configures the interface type and enters interface configuration mode.
Step 4	**ip rsvp bandwidth** [*interface-kbps*] [*single-flow-kbps*] **Example:** `Device(config-if)# ip rsvp bandwidth`	Enables RSVP on an interface. • The optional *interface-kbps* and *single-flow-kbps* arguments specify the amount of bandwidth that can be allocated by RSVP flows or to a single flow, respectively. Values are from 1 to 10000000. **Note** Repeat this command for each interface on which you want to enable RSVP.
Step 5	**ipv6 rsvp source** {**address** *ipv6-address* | **interface** *type number*} **Example:** `Device(config-if)# ipv6 rsvp source address 2001:DB8::1`	Configures an RSVP device to populate an address other than the native interface address in the PHOP address field of the hop object when forwarding a PATH message onto that interface. **Note** The source IPv6 address that you configure should be a valid local IP address.
Step 6	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.

Configuration Examples for CAC for IPv6 Flows

- Example: Entering Senders or Receivers for IPv6 Flows to the RSVP Database, page 12
- Example: Configuring a Static Sender for IPv6 Flows, page 12
- Example: Configuring a Static Receiver for IPv6 Flows, page 12
- Example: Configuring a Receiver Proxy for IPv6 Flows on a Tailend Device, page 12
- Example: Configuring RSVP as a Transport Protocol for IPv6 Flows, page 13
- Example: Binding a Key Chain to an RSVP IPv6 Neighbor, page 13
- Example: Configuring PHOP for IPv6 Flows, page 13

Example: Entering Senders or Receivers for IPv6 Flows to the RSVP Database

The following example shows how to add senders or receivers for IPv6 flows to the RSVP database:

```
Device> enable
Device# configure terminal
Device(config)# ipv6 rsvp sender 2001:DB8:1:: 2001:DB8:2:: tcp 2 3 2001:DB8::1
fastEthernet 0/1 2 3
Device(config)# ipv6 rsvp reservation 2001:DB8:1:: 2001:DB8::2 tcp 2 3 2001:DB8::3
fastEthernet 0/1 ff load 2 4
Device(config)# end
```

Example: Configuring a Static Sender for IPv6 Flows

The following example shows how to configure a static sender for IPv6 flows:

```
Device> enable
Device# configure terminal
Device(config)# ipv6 rsvp sender-host 2001:DB8:1:: 2001:DB8:2:: udp 1 1 10 10 vrf myvrf
Device(config)# end
```

Example: Configuring a Static Receiver for IPv6 Flows

The following example shows how to configure a static receiver for IPv6 flows:

```
Device> enable
Device# configure terminal
Device(config)# ipv6 rsvp reservation-host 2001:DB8::1 2001:DB8::2 udp 20 30 se load 100
60 identity rsvp-voice
Device(config)# end
```

Example: Configuring a Receiver Proxy for IPv6 Flows on a Tailend Device

The following example shows how to configure a receiver proxy for IPv6 flows on a tailend device:

```
Device> enable
Device# configure terminal
Device(config)# ipv6 rsvp listener vrf myvrf 2001:DB8:1:: any any reply
Device(config)# end
```

Example: Configuring RSVP as a Transport Protocol for IPv6 Flows

The following example shows how to configure RSVP as transport protoocol for IPv6 flows:

```
Device> enable
Device# configure terminal
Device(config)# ipv6 rsvp transport client 2
Device(config)# ipv6 rsvp transport sender-host tcp 2001:DB8:10:: 2001:DB:11:: 3 4 5 2 3
4 vrf vrf1
Device(config)# end
```

Example: Binding a Key Chain to an RSVP IPv6 Neighbor

The following example shows how to bind a key chain to an RSVP IPv6 neighbor:

```
Device> enable
Device# configure terminal
Device(config)# ipv6 rsvp authentication neighbor access-list 1 key-chain neighbor_V
Device(config)# end
```

Example: Configuring PHOP for IPv6 Flows

The following example shows how to configure PHOP for IPv6 flows:

```
Device# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)# interface ethernet 1/0
Device(config-if)# ipv6 rsvp source address 2001:DB8::1
Device(config-if)# end
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	*Master Commands List, All Releases*
RSVP commands	*Quality of Service Command Reference*
Configuring RSVP, RSVP-PHOP, RSVP Fast Local Repair, RSVP Message Authentication and other related RSVP features	*QoS: RSVP Configuration Guide*

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
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.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for CAC for IPv6 Flows

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 CAC for IPv6 Flows

Feature Name	Releases	Feature Information
CAC for IPv6 Flows	15.2(3)T	The CAC for IPv6 Flows feature provides IPv6 support for RSVP. By enabling this feature, the network is made to support the complete RSVP IPv6 functionality for CAC and Medianet. The following commands were introduced or modified: **clear ipv6 rsvp authentication**, **clear ipv6 rsvp reservation**, **clear ipv6 rsvp sender**, **ip rsvp bandwidth**, **ip rsvp listener outbound**, **ip rsvp signaling rate-limit**, **ipv6 rsvp sender**, **ipv6 rsvp sender-host**, **ipv6 rsvp source**, **ipv6 rsvp listener**, **ipv6 rsvp reservation**, **ipv6 rsvp reservation-host**, **ipv6 rsvp transport sender-host**, **show ipv6 rsvp authentication**, **show ipv6 rsvp host**, **show ipv6 rsvp installed**, **show ipv6 rsvp listeners**, **show ipv6 rsvp neighbor**, **show ipv6 rsvp request**, **show ipv6 signaling blockade**, **show ipv6 rsvp transport sender**.