

ip header-compression disable-feedback

To disable the CONTEXT_STATUS feedback messages from the interface or link, use the **ip header-compression disable-feedback** command in interface configuration mode. To enable CONTEXT_STATUS feedback messages from the interface or link, use the **no** form of this command.

ip header-compression disable-feedback

no ip header-compression disable-feedback

Syntax Description

This command has no arguments or keywords.

Defaults

CONTEXT_STATUS feedback messages are enabled by default.

Command Modes

Interface configuration

Command History

Release	Modification
12.3(2)T	This command was introduced.

Usage Guidelines

The **ip header-compression disable-feedback** command is designed for use with satellite links where the path for the upward link is different from the path for the downward link. When the paths are different, CONTEXT_STATUS messages are not useful.

The **ip header-compression disable-feedback** command can be used with either Real-Time Transport Protocol (RTP) or TCP header compression.

Examples

The following example disables the CONTEXT_STATUS messages on the Serial2/0.1 subinterface:

```
Router> enable
Router# configure terminal
Router(config-if)# interface Serial2/0.1
Router(config-if)# ip header-compression disable-feedback
Router(config-if)# exit
```

Related Commands

Command	Description
ip header-compression max-header	Specifies the maximum size of the compressed IP header.
ip header-compression max-period	Specifies the maximum number of compressed packets between full headers.
ip header-compression max-time	Specifies the maximum amount of time to wait before refreshing the compressed IP header.

ip header-compression max-header

To specify the maximum amount of time to wait before the compressed IP header is refreshed, use the **ip header-compression max-header** command in interface configuration mode. To return the amount of time to wait before the compressed IP header is refreshed to the default value, use the **no** form of this command.

ip header-compression max-header *max-header-size*

no ip header-compression max-header

Syntax Description	<i>max-header-size</i> Size of the IP header, in bytes. The size of the IP header can be in the range of 20 to 168 bytes.
---------------------------	---

Defaults	168 bytes
-----------------	-----------

Command Modes	Interface configuration
----------------------	-------------------------

Command History	Release	Modification
	12.3(2)T	This command was introduced.

Usage Guidelines	The <i>max-header-size</i> argument of the ip header-compression max-header command can be used to restrict the size of the header to be compressed.
-------------------------	---

Examples	In the following example, the ip header-compression max-header command is configured to specify the maximum IP header size of the packet. In this configuration, the maximum IP header size is 100 bytes.
-----------------	--

```
Router> enable
Router# configure terminal
Router(config-if)# interface Serial2/0.1
Router(config-if)# ip header-compression max-header 100
Router(config-if)# exit
```

Related Commands	Command	Description
	ip header-compression disable-feedback	Disables CONTEXT_STATUS feedback messages from the interface or link.
	ip header-compression max-period	Specifies the maximum number of compressed packets between full headers.
	ip header-compression max-time	Specifies the maximum amount of time to wait before refreshing the compressed IP header.

ip header-compression max-period

To specify the maximum number of compressed packets between full headers, use the **ip header-compression max-period** command in interface configuration mode. To return the number of compressed packets to the default value, use the **no** form of this command.

ip header-compression max-period *number-of-packets*

no ip header-compression max-period

Syntax Description	<i>number-of-packets</i> Specifies a number of packets between full headers. The number can be in the range of 0 to 65535 packets.
---------------------------	--

Defaults	256 packets
-----------------	-------------

Command Modes	Interface configuration
----------------------	-------------------------

Command History	Release	Modification
	12.3(2)T	This command was introduced.

Usage Guidelines	<p>With the ip header-compression max-period command, full IP packet headers are sent in an exponentially increasing period after there has been a change in the context status. This exponential increase in the time period avoids the necessity of exchanging messages between the mechanism compressing the header and the mechanism decompressing the header.</p> <p>By default, the ip header-compression max-period command operates on User Datagram Protocol (UDP) traffic only. However, if the periodic refresh keyword of either the frame-relay ip rtp header-compression command or the frame-relay map ip rtp header-compression command is configured, the ip header-compression max-period command operates on both UDP and Real-Time Transport Protocol (RTP) traffic.</p>
-------------------------	--

Examples	<p>In the following example, the ip header-compression max-period command is configured to specify the number of packets between full header packets. In this configuration, the packet number specified is 160.</p>
-----------------	---

```
Router> enable
Router# configure terminal
Router(config-if)# interface Serial12/0.1
Router(config-if)# ip header-compression max-period 160
Router(config-if)# exit
```

Related Commands	Command	Description
	frame-relay ip rtp header-compression	Enables RTP header compression for all Frame Relay maps on a physical interface.
	frame-relay map ip rtp header-compression	Enables RTP header compression per DLCI.
	ip header-compression disable-feedback	Disables CONTEXT_STATUS feedback messages from the interface or link.
	ip header-compression max-header	Specifies the maximum size of the compressed IP header.
	ip header-compression max-time	Specifies the maximum amount of time to wait before refreshing the compressed IP header.

ip header-compression max-time

To specify the maximum amount of time to wait before refreshing the compressed IP header, use the **ip header-compression max-time** command in interface configuration mode. To return to the default value, use the **no** form of this command.

ip header-compression max-time *length-of-time*

no ip header-compression max-time

Syntax Description	<i>length-of-time</i>	Specifies a different amount of time (other than the default) to wait before refreshing the IP header. The amount of time can be in the range of 0 to 65535 seconds.
---------------------------	-----------------------	--

Defaults	5 seconds
-----------------	-----------

Command Modes	Interface configuration
----------------------	-------------------------

Command History	Release	Modification
	12.3(2)T	This command was introduced.

Usage Guidelines	<p>The ip header-compression max-time command is designed to avoid losing too many packets if the context status of the receiver has been lost.</p> <p>If a packet is to be sent and the maximum amount of time has elapsed since the last time the IP header was refreshed, a full header is sent.</p> <p>By default, the ip header-compression max-time command operates on User Datagram Protocol (UDP) traffic only. However, if the periodic refresh keyword of either the frame-relay ip rtp header-compression command or the frame-relay map ip rtp header-compression command is configured, the ip header-compression max-time command operates on UDP and Real-Time Transport Protocol (RTP) traffic.</p>
-------------------------	--

Examples	<p>In the following example, the ip header-compression max-time command is configured to specify the maximum amount of time to wait before refreshing the compressed IP header. In this configuration the amount of time to wait is 30 seconds.</p>
-----------------	--

```
Router> enable
Router# configure terminal
Router(config-if)# interface Serial12/0.1
Router(config-if)# ip header-compression max-time 30
Router(config-if)# exit
```

Related Commands	Command	Description
	frame-relay ip rtp header-compression	Enables RTP header compression for all Frame Relay maps on a physical interface.
	frame-relay map ip rtp header-compression	Enables RTP header compression per DLCI.
	ip header-compression disable-feedback	Disables CONTEXT_STATUS feedback messages from the interface or link.
	ip header-compression max-header	Specifies the maximum size of the compressed IP header.
	ip header-compression max-period	Specifies the maximum number of compressed packets between full headers.

ip header-compression recoverable-loss

To enable Enhanced Compressed Real-Time Transport Protocol (ECRTP) on an interface, use the **ip header-compression recoverable-loss** command in interface configuration mode. To disable ECRTP on an interface, use the **no** form of this command.

ip header-compression recoverable-loss {dynamic | *n*}

no ip header-compression recoverable-loss

Syntax Description	dynamic	Dynamic recoverable loss calculation.
	<i>n</i>	Maximum number of consecutive packet drops. Ranges from 1 to 8.

Defaults When using the keyword **dynamic**, the default value is 4.

Command Modes Interface configuration

Command History	Release	Modification
	12.3(11)T	This command was introduced.

Usage Guidelines Enhanced CRTP reduces corruption by changing the way the compressor updates the context at the decompressor. The compressor sends changes multiple times to keep the compressor and decompressor synchronized. This method is characterized by a number *n* that represents the quality of the link between the hosts. By repeating the updates, the probability of context corruption due to packet loss is minimized. The *n* value is maintained independently for each context and is not required to be the same for all contexts.

Examples In the following example, a serial interface is configured with Point-to-Point Protocol (PPP) encapsulation, and ECRTP is enabled with dynamic loss recovery:

```
Router(config)# interface serial 2/0
Router(config-if)# encapsulation ppp
Router(config-if)# ip rtp header-compression ietf-format
Router(config-if)# ip header-compression recoverable-loss dynamic
Router(config-if)# end
```

Related Commands	Command	Description
	debug ip rtp error	Displays RTP header compression errors.
	debug ip rtp header-compression	Displays events specific to RTP header compression.
	ip rtp header-compression	Enables RTP header compression.
	show ip rtp header-compression	Displays RTP header compression statistics.

ip nbar custom

To extend the capability of network-based application recognition (NBAR) Protocol Discovery to classify and monitor additional static port applications or to allow NBAR to classify nonsupported static port traffic, use the **ip nbar custom** command in global configuration mode.

```
ip nbar custom name [offset [format value] {variable field-name field-length}]
[source | destination] [tcp | udp] [range start end | port-number]
```

```
no ip nbar custom name [offset [format value] {variable field-name field-length}]
[source | destination] [tcp | udp] [range start end | port-number]
```

Syntax Description	
<i>name</i>	The name given to the custom protocol. This name is reflected wherever the name is used, including NBAR Protocol Discovery, the match protocol command, the ip nbar port-map command, and the NBAR Protocol Discovery MIB. The name must be no longer than 24 characters and can only contain uppercase and lowercase letters, digits, and the underscore (_) character.
<i>offset</i>	(Optional) A digit representing the byte location for payload inspection. The offset function is based on the beginning of the payload directly after the TCP or User Datagram Protocol (UDP) header.
<i>format</i>	(Optional) Defines the format of the value that is being inspected in the packet payload. Current options are ascii , hex , and decimal .
<i>value</i>	(Optional) The value being searched in the packet inspection. The length of the value is dependant on the chosen <i>format</i> . The length restrictions for each format are listed below: <ul style="list-style-type: none"> • ascii—up to 16 characters can be searched. Regular expressions are not supported. • hex—up to 4 bytes. • decimal—up to 4 bytes.
variable <i>field-name field-length</i>	When the variable keyword is entered, a specific portion of the custom protocol can be treated as an NBAR-supported protocol (for example, a specific portion of the custom protocol can be tracked using class-map statistics and can be matched using the class-map command). If the variable keyword is entered, the following fields must be defined: <ul style="list-style-type: none"> • <i>field-name</i>—Provides a name for the field to search in the payload. After a custom protocol is configured using a variable, this <i>field-name</i> can be used with up to 24 different values per router configuration. • <i>field-length</i>—Enters the field length in bytes. The field length can be up to 4 bytes, so the <i>field-length</i> value can be entered as 1, 2, 3, or 4.
<i>source</i> <i>destination</i>	(Optional) Specifies the direction in which packets are inspected. If source or destination is not specified, all packets traveling in either direction are monitored by NBAR.
tcp udp	(Optional) Specifies the TCP or the UDP implemented by the application.

range <i>start end</i>	(Optional) Specifies a range of ports that the custom application monitors. The start is the first port in the range, and the end is the last port in the range. One range of up to 1000 ports can be specified for each custom protocol.
<i>port-number</i>	(Optional) The port that the custom application monitors. Up to 16 individual ports can be specified as a single custom protocol.

Defaults

If source or destination is not specified, traffic flowing in both directions is inspected if the custom protocol is enabled in NBAR.

Command Modes

Global configuration

Command History

Release	Modification
12.3(4)T	This command was introduced.
12.3(11)T	The variable keyword was introduced.

Usage Guidelines

More than 30 custom applications can be created on the router.

NBAR can support up to 128 protocols total.

If the **variable** keyword is entered while configuring the custom protocol, traffic statistics for the variable appear in some NBAR class map **show** outputs.

Up to 24 variable values per custom protocol can be expressed in class maps. For instance, in the following configuration, 4 variables are used and 20 more “scid” values could be used.

```
ip nbar custom ftdd 125 variable scid 1 tcp range 5001 5005

class-map match-any active-craft
match protocol ftdd scid 0x15
match protocol ftdd scid 0x21

class-map match-any passive-craft
match protocol ftdd scid 0x11
match protocol ftdd scid 0x22
```

Examples

In the following example, the custom protocol “app_sales1” identifies TCP packets with a source port of 4567 and contains the term “SALES” in the fifth byte of the payload:

```
ip nbar custom app_sales1 5 ascii SALES source tcp 4567
```

In the following example, the custom protocol “virus_home” identifies UDP packets with a destination port of 3000 and contains “0x56” in the seventh byte of the payload:

```
ip nbar custom virus_home 7 hex 0x56 dest udp 3000
```

In the following example, custom protocol “media_new” identifies TCP packets with a destination or source port of 4500 and have a value of 90 in the sixth byte of the payload:

```
ip nbar custom media_new 6 decimal 90 tcp 4500
```

In the following example, custom protocol “msn1” looks for TCP packets with a destination or source port of 6700:

```
ip nbar custom msn1 tcp 6700
```

In the following example, custom protocol “mail_x” looks for UDP packets with a destination port of 8202.

```
ip nbar custom mail_x destination udp 8202
```

In the following example, custom protocol “mail_y” looks for UDP packets with destination ports between 3000 and 4000 including 3000 and 4000 as well as port 5500:

```
ip nbar custom mail_y destination udp range 3000 4000 5500
```

In the following example, custom protocol “ftdd” is created using a variable. A class map matching this custom protocol based on the variable is also created. In this example, class map “matchscidinftdd” matches all traffic entering or leaving TCP ports 5001-5005 that has the value “804” at byte 125. The variable scid is 2 bytes in length.

```
ip nbar custom ftdd 125 variable scid 2 tcp range 5001 5005
```

```
class-map matchscidinftdd  
match protocol ftdd scid 804
```

The same example above can also be done using hexadecimal values in the class map as follows:

```
ip nbar custom ftdd 125 variable scid 2 tcp range 5001 5005
```

```
class-map matchscidinftdd  
match protocol ftdd scid 0x324
```

In the following example, the **variable** keyword is used while creating a custom protocol, and class maps are configured to classify different values within the variable field into different traffic classes. Specifically, in the example below, variable scid values 0x15, 0x21, and 0x27 are classified into class map “active-craft” while scid values 0x11, 0x22, and 0x25 are classified into class map “passive-craft.”

```
ip nbar custom ftdd 125 variable scid 1 tcp range 5001 5005
```

```
class-map match-any active-craft  
match protocol ftdd scid 0x15  
match protocol ftdd scid 0x21  
match protocol ftdd scid 0x27
```

```
class-map match-any passive-craft  
match protocol ftdd scid 0x11  
match protocol ftdd scid 0x22  
match protocol ftdd scid 0x25
```

ip nbar pdlm

To extend or enhance the list of protocols recognized by network-based application recognition (NBAR) through a Cisco-provided Packet Description Language Module (PDLM), use the **ip nbar pdlm** command in global configuration mode. To unload a PDLM if it was previously loaded, use the **no** form of this command.

ip nbar pdlm *pdlm-name*

no ip nbar pdlm *pdlm-name*

Syntax Description

<i>pdlm-name</i>	URL at which the PDLM can be found on the Flash card.
------------------	---

Defaults

No default behavior or values

Command Modes

Global configuration

Command History

Release	Modification
12.0(5)XE2	This command was introduced.
12.1(1)E	This command was integrated into Cisco IOS Release 12.1E.
12.1(5)T	This command was integrated into Cisco IOS Release 12.1T.
12.1(13)E	This command was implemented on Catalyst 6000 family switches without FlexWAN modules.

Usage Guidelines

This command is used to extend the list of protocols recognized by a given version of NBAR or to enhance an existing protocol recognition capability. NBAR can be given an external PDLM at run time. In most cases, the PDLM enables NBAR to recognize new protocols without requiring a new Cisco IOS image or a router reload. Only Cisco can provide you with a new PDLM.

A list of the available PDLMs can be viewed online at Cisco.com.

Examples

The following example configures NBAR to load the citrix.pdlm PDLM from Flash memory on the router:

```
ip nbar pdlm flash://citrix.pdlm
```

Related Commands

Command	Description
show ip nbar pdlm	Displays the current PDLM in use by NBAR.

ip nbar port-map

To configure network-based application recognition (NBAR) to search for a protocol or protocol name using a port number other than the well-known port, use the **ip nbar port-map** command in global configuration mode. To look for the protocol name using only the well-known port number, use the **no** form of this command.

ip nbar port-map *protocol-name* [**tcp** | **udp**] *port-number*

no ip nbar port-map *protocol-name* [**tcp** | **udp**] *port-number*

Syntax Description	
<i>protocol-name</i>	Name of protocol known to NBAR.
tcp	(Optional) Specifies that a TCP port will be searched for the specified <i>protocol-name</i> argument.
udp	(Optional) Specifies that a User Datagram Protocol (UDP) port will be searched for the specified <i>protocol-name</i> argument.
<i>port-number</i>	Assigned port for named protocol. The <i>port-number</i> argument is either a UDP or a TCP port number, depending on which protocol is specified in this command line. Up to 16 <i>port-number</i> arguments can be specified in one command line. Port number values can range from 0 to 65535.

Defaults Disabled

Command Modes Global configuration

Command History	Release	Modification
	12.0(5)XE2	This command was introduced.
	12.1(1)E	This command was integrated into Cisco IOS Release 12.1(1)E.
	12.1(5)T	This command was integrated into Cisco IOS Release 12.1(5)T.
	12.1(13)E	This command was implemented on Catalyst 6000 family switches without FlexWAN modules.

Usage Guidelines This command is used to tell NBAR to look for the protocol or protocol name, using a port number or numbers other than the well-known Internet Assigned Numbers Authority (IANA)-assigned port number. For example, use this command to configure NBAR to look for Telnet on a port other than 23. Up to 16 ports can be specified with this command. Port number values can range from 0 to 65535.

Examples

The following example configures NBAR to look for the protocol Structured Query Language (SQL)*NET on port numbers 63000 and 63001 instead of on the well-known port number:

```
ip nbar port-map sqlnet tcp 63000 63001
```

Related Commands

Command	Description
show ip nbar port-map	Displays the current protocol-to-port mappings in use by NBAR.

ip nbar protocol-discovery

To configure networked-based application recognition (NBAR) to discover traffic for all protocols known to NBAR on a particular interface, use the **ip nbar protocol-discovery** command in interface configuration mode. To disable traffic discovery, use the **no** form of this command.

ip nbar protocol-discovery

no ip nbar protocol-discovery

Syntax Description This command has no arguments or keywords.

Defaults Disabled

Command Modes Interface configuration

Command History	Release	Modification
	12.0(5)XE2	This command was introduced.
	12.1(1)E	This command was integrated into Cisco IOS Release 12.1E.
	12.1(5)T	This command was integrated into Cisco IOS Release 12.1T.
	12.1(13)E	This command was implemented on Catalyst 6000 family switches without FlexWAN modules.

Usage Guidelines Use the **ip nbar protocol-discovery** command to configure NBAR to keep traffic statistics for all protocols known to NBAR. Protocol discovery provides an easy way to discover application protocols transiting an interface so that QoS policies can be developed and applied. The Protocol Discovery feature discovers any protocol traffic supported by NBAR. Protocol discovery can be used to monitor both input and output traffic and may be applied with or without a service policy enabled.

Examples The following example configures protocol discovery on an Ethernet interface:

```
interface ethernet 1/3
 ip nbar protocol-discovery
```

Related Commands	Command	Description
	show ip nbar protocol-discovery	Displays the statistics gathered by the NBAR Protocol Discovery feature.

ip rsvp admission-control compression predict

To configure Resource Reservation Protocol (RSVP) admission control compression prediction, use the **ip rsvp admission-control compression predict** command in interface configuration mode. To disable compression prediction, use the **no** form of this command.

```
ip rsvp admission-control compression predict [method { rtp | udp } [bytes-saved N]]
```

```
no ip rsvp admission-control compression predict [method { rtp | udp } [bytes-saved N]]
```

Syntax Description	method	(Optional) Type of compression used.
	rtp udp	Real-Time Transport Protocol (RTP) or User Data Protocol (UDP) compression schemes.
	bytes-saved <i>N</i>	(Optional) Predicted number of bytes saved per packet when RSVP predicts that compression will occur using the specified method. Values for <i>N</i> for RTP are 1 to 38; for UDP, 1 to 26.

Defaults This command is enabled by default. The default value of bytes saved for RTP is 36; for UDP, 20.

Command Modes Interface configuration

Command History	Release	Modification
	12.2(15)T	This command was introduced.

Usage Guidelines Use the **ip rsvp admission-control compression predict** command to disable or enable the RSVP prediction of compression for a specified method or all methods if neither **rtp** nor **udp** is selected. You can adjust the default compressibility parameter that RSVP uses to compute the compression factor for each flow.

If you use the **ip rsvp admission-control compression predict** command to change the compression method or the number of bytes saved per packet, these values affect only new flows, not existing ones.

There are two approaches to compression—conservative and aggressive. When you predict compression conservatively, you assume savings of fewer bytes per packet, but receive a higher likelihood of guaranteed quality of service (QoS). You are allowed more bandwidth per call, but each link accommodates fewer calls. When you predict compression aggressively, you assume savings of more bytes per packet, but receive a lower likelihood of guaranteed QoS. You are allowed less bandwidth per call, but each link accommodates more calls.

Examples The following command sets the compressibility parameter for flows using the RTP method to 30 bytes saved per packet:

```
Router(config-if)# ip rsvp admission-control compression predict method rtp bytes-saved 30
```

The following command sets the compressibility parameter for flows using the UDP method to 20 bytes saved per packet:

```
Router(config-if)# ip rsvp admission-control compression predict method udp bytes-saved 20
```

The following command disables RTP header compression prediction:

```
Router(config-if)# no ip rsvp admission-control compression predict method rtp
```

The following command disables UDP header compression prediction:

```
Router(config-if)# no ip rsvp admission-control compression predict method udp
```

**Note**

Disabling the compressibility parameter affects only those flows using the specified method.

Related Commands

Command	Description
show ip rtp header-compression	Displays statistics about RTP header compression.

ip rsvp atm-peak-rate-limit

To set a limit on the peak cell rate (PCR) of reservations for all newly created Resource Reservation Protocol (RSVP) switched virtual circuits (SVCs) established on the current interface or any of its subinterfaces, use the **ip rsvp atm-peak-rate-limit** command in interface configuration mode. To remove the current peak rate limit, in which case the reservation peak rate is limited by the line rate, use the **no** form of this command.

ip rsvp atm-peak-rate-limit *limit*

no ip rsvp atm-peak-rate-limit

Syntax Description	<i>limit</i>	The peak rate limit of the reservation specified, in KB. The minimum value allowed is 1 KB; the maximum value allowed is 2 GB.
---------------------------	--------------	--

Defaults	The peak rate of a reservation defaults to the line rate.
-----------------	---

Command Modes	Interface configuration
----------------------	-------------------------

Command History	Release	Modification
	12.0(3)T	This command was introduced.

Usage Guidelines	Each RSVP reservation corresponds to an ATM SVC with a certain peak cell rate (PCR), sustainable cell rate (SCR), and maximum burst size. The PCR, also referred to as the peak rate, can be configured by the user or allowed to default to the line rate.
-------------------------	---

RSVP controlled-load reservations do not define any peak rate for the data. By convention, the allowable peak rate in such reservations is taken to be infinity, which is usually represented by a very large number. Under these circumstances, when a controlled-load reservation is converted to an ATM SVC, the peak cell rate for the SVC becomes correspondingly large and may be out of range for the switch. You can use the **ip rsvp atm-peak-rate-limit** command to limit the peak rate.

The following conditions determine the peak rate limit on the RSVP SVC:

- The peak rate defaults to the line rate.
- If the peak rate is greater than the configured peak rate limiter, the peak rate is lowered to the peak rate limiter.
- The peak rate cannot be less than the reservation bandwidth. If this is the case, the peak rate is raised to the reservation bandwidth.



Note

Bandwidth conversions applied to the ATM space from the RSVP space are also applied to the peak rate.

The peak rate limit is local to the router; it does not affect the normal messaging of RSVP. Only the SVC setup is affected. Large peak rates are sent to the next host without modification.

For RSVP SVCs established on subinterfaces, the peak rate limit applied to the subinterface takes effect on all SVCs created on that subinterface. If a peak rate limit is applied to the main interface, the rate limit has no effect on SVCs created on a subinterface of the main interface even if the limit value on the main interface is lower than the limit applied to the subinterface.

For a given interface or subinterface, a peak rate limit applied to that interface affects only new SVCs created on the interface, not existing SVCs.

**Note**

This command is available only on interfaces that support the **ip rsvp svc-required** command.

Use the **show ip rsvp atm-peak-rate-limit** command to determine the peak rate limit set for an interface or subinterface, if one is configured.

Examples

The following example sets the peak rate limit (PCR limit) for interface atm2/0/0.1 to 100 KB:

```
interface atm2/0/0.1
 ip rsvp atm-peak-rate-limit 100
```

Related Commands

Command	Description
ip rsvp svc-required	Enables creation of an SVC to service any new RSVP reservation made on the interface or subinterface.
show ip rsvp interface	Displays RSVP-related interface information.

ip rsvp authentication

To activate Resource Reservation Protocol (RSVP) cryptographic authentication, use the **ip rsvp authentication** command in interface configuration mode. To deactivate authentication, use the **no** form of this command.

ip rsvp authentication

no ip rsvp authentication

Syntax Description This command has no arguments or keywords.

Defaults This command is disabled by default.

Command Modes Interface configuration

Release	Modification
12.2(15)T	This command was introduced.

Usage Guidelines Use the **ip rsvp authentication** command to deactivate and then reactivate RSVP authentication without reentering the other RSVP authentication configuration commands. You should not enable authentication unless you have previously configured a key. If you issue this command before the **ip rsvp authentication key** command, you get a warning message indicating that RSVP discards all messages until you specify a key. The **no ip rsvp authentication** command disables RSVP cryptographic authentication. However, the command does not automatically remove any other authentication parameters that you have configured. You must issue a specific **no ip rsvp authentication** command; for example, **no ip rsvp authentication key**, **no ip rsvp authentication type**, or **no ip rsvp authentication window-size**, if you want to remove them from the configuration.

The **ip rsvp authentication** command is similar to the **ip rsvp neighbor** command. However, the **ip rsvp authentication** command provides better authentication and performs system logging.

Examples The following command activates authentication on an interface:

```
Router(config-if)# ip rsvp authentication
```

The following command deactivates authentication on an interface:

```
Router(config-if)# no ip rsvp authentication
```

Related Commands	Command	Description
	ip rsvp authentication key	Specifies the key (string) for the RSVP authentication algorithm.
	ip rsvp authentication type	Specifies the algorithm used to generate cryptographic signatures in RSVP messages.
	ip rsvp authentication window-size	Specifies the maximum number of Resource Reservation Protocol (RSVP) authenticated messages that can be received out of order
	ip rsvp neighbor	Enables neighbors to request a reservation.

ip rsvp authentication challenge

To make Resource Reservation Protocol (RSVP) perform a challenge-response handshake with any new RSVP neighbors on a network, use the **ip rsvp authentication challenge** command in interface configuration mode. To disable the challenge-response handshake, use the **no** form of this command.

ip rsvp authentication challenge

no ip rsvp authentication challenge

Syntax Description

This command has no arguments or keywords.

Defaults

This command is disabled by default.

Command Modes

Interface configuration

Command History

Release	Modification
12.2(15)T	This command was introduced.

Usage Guidelines

The **ip rsvp authentication challenge** command requires RSVP to perform a challenge-response handshake with any new RSVP neighbors that are discovered on a network. Such a handshake allows the router to thwart RSVP message replay attacks while booting, especially if there is a long period of inactivity from trusted RSVP neighbors following the reboot. If messages from trusted RSVP neighbors arrive very quickly after the router reboots, then challenges may not be required because the router will have reestablished its security associations with the trusted nodes before the untrusted nodes can attempt replay attacks.

If you enable RSVP authentication challenges, you should consider enabling RSVP refresh reduction by using the **ip rsvp signalling refresh reduction** command. While a challenge handshake is in progress, the receiving router initiating the handshake discards all RSVP messages from the node being challenged until the handshake-initiating router receives a valid challenge response.



Note

If a neighbor does not reply to the first challenge message after 1 second, Cisco IOS sends another challenge message and waits 2 seconds. If no response is received to the second challenge, Cisco IOS sends another and waits 4 seconds. If no response to the third challenge is received, Cisco IOS sends a fourth challenge and waits 8 seconds. If there is no response to the fourth challenge, Cisco IOS stops the current challenge to that neighbor, logs a system error message, and does not create a security association for that neighbor. This kind of exponential backoff is used to recover from challenges dropped by the network or busy neighbors.

Activating refresh reduction enables the challenged node to resend dropped messages more quickly once the handshake has completed. This causes RSVP to reestablish reservation state faster when the router reboots.

Enable authentication challenges wherever possible to reduce the router's vulnerability to replay attacks.

Examples

The following command shows how to enable RSVP to perform a challenge-response handshake:

```
Router(config-if)# ip rsvp authentication challenge
```

Related Commands

Command	Description
ip rsvp signalling refresh reduction	Enables RSVP refresh reduction.

ip rsvp authentication key

To specify the key (string) for the Resource Reservation Protocol (RSVP) authentication algorithm, use the **ip rsvp authentication key** command in interface configuration mode. To disable the key, use the **no** form of this command.

ip rsvp authentication key *passphrase*

no ip rsvp authentication key

Syntax Description	<i>passphrase</i>	Phrase that ranges from 8 to 40 characters. See “Usage Guidelines” for additional information.
Defaults	No key is specified.	
Command Modes	Interface configuration	
Command History	Release	Modification
	12.2(15)T	This command was introduced.

Usage Guidelines

Use the **ip rsvp authentication key** command to select the key for the authentication algorithm. This key is a passphrase of 8 to 40 characters. It can include spaces; quotes are not required if spaces are used. The key can consist of more than one word. We recommend that you make the passphrase as long as possible. This key must be the same for all RSVP neighbors on this interface. As with all passwords, you should choose them carefully so that attackers cannot easily guess them.

Here are some guidelines:

- Use a mixture of upper- and lowercase letters, digits, and punctuation.
- If using just a single word, do not use a word contained in any dictionary of any language, spelling lists, or other lists of words.
- Use something easily remembered so you do not have to write it down.
- Do not let it appear in clear text in any file or script or on a piece of paper attached to a terminal.

By default, RSVP authentication keys are stored in clear text in the router configuration file, but they can optionally be stored as encrypted text in the configuration file. To enable key encryption, use the global configuration **key config-key 1** *string* command. After you enter this command, the passphrase parameter of each **ip rsvp authentication key** command is encrypted with the Data Encryption Standard (DES) algorithm when you save the configuration file. If you later issue a **no key config-key 1** *string* command, the RSVP authentication key is stored in clear text again when you save the configuration.

The *string* argument is not stored in the configuration file; it is stored only in the router's private NVRAM and will not appear in the output of a **show run** or **show config** command. Therefore, if you copy the configuration file to another router, any encrypted RSVP keys in that file will not be successfully decrypted by RSVP when the router boots and RSVP authentication will not operate correctly. To recover from this, follow these steps on the new router:

1. For each RSVP interface with an authentication key, issue a **no ip rsvp authentication key** command to clear the old key.
2. For that same set of RSVP interfaces, issue an **ip rsvp authentication key** command to reconfigure the correct clear text keys.
3. Issue a global **key config-key 1 string** command to reencrypt the RSVP keys for the new router.
4. Save the configuration.

Examples

The following command sets the passphrase to 11223344 in clear text:

```
Router(config-if)# ip rsvp authentication key 11223344
```

To encrypt the authentication key, issue the **key config-key 1 string** command as follows:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# key config-key 1 11223344
Router(config)# end
```

Related Commands

Command	Description
key config-key	Defines a private DEF key for the router.

ip rsvp authentication lifetime

To control how long Resource Reservation Protocol (RSVP) maintains security associations with other trusted RSVP neighbors, use the **ip rsvp authentication lifetime** command in interface configuration mode. To disable the lifetime setting, use the **no** form of this command.

ip rsvp authentication lifetime *hh:mm:ss*

no ip rsvp authentication lifetime *hh:mm:ss*

Syntax Description	<i>hh:mm:ss</i>	Hours: minutes: seconds that RSVP maintains security associations with other trusted RSVP neighbors. The range is 1 second to 24 hours. The default is 30 minutes.
---------------------------	-----------------	--

Defaults	Default security association is 30 minutes; range is 1 second to 24 hours.
-----------------	--

Command Modes	Interface configuration
----------------------	-------------------------

Command History	Release	Modification
	12.2(15)T	This command was introduced.

Usage Guidelines	<p>Use the ip rsvp authentication lifetime command to indicate when to end security associations with RSVP trusted neighbors. If an association's lifetime expires, but at least one valid, RSVP authenticated message was received in that time period, RSVP resets the security association's lifetime to this configured value. When a neighbor stops sending RSVP signaling messages (that is, the last reservation has been torn down), the memory used for the security association is freed as well as when the association's lifetime period ends. The association can be re-created if that RSVP neighbor resumes its signaling. Setting the lifetime to shorter periods allows memory to be recovered faster when the router is handling a lot of short-lived reservations. Setting the lifetime to longer periods reduces the workload on the router when establishing new authenticated reservations.</p>
-------------------------	--

Use the **clear ip rsvp authentication** command to free security associations before their lifetimes expire.

Examples	The following command sets the lifetime period for 30 minutes and 5 seconds:
-----------------	--

```
Router(config-if)# ip rsvp authentication lifetime 00:30:05
```

Related Commands	Command	Description
	clear ip rsvp authentication	Eliminates RSVP security associations before their lifetimes expire.

ip rsvp authentication type

To specify the algorithm used to generate cryptographic signatures in Resource Reservation Protocol (RSVP) messages, use the **ip rsvp authentication type** command in interface configuration mode. To disable the type (or to use the default type, **md5**), use the **no** form of this command.

ip rsvp authentication type { md5 | sha-1 }

no ip rsvp authentication type

Syntax Description

md5	RSA Message Digest 5 algorithm.
sha-1	National Institute of Standards and Technologies (NIST) Secure Hash Algorithm-1; it is newer and more secure than MD5.

Defaults

The default type is **md5**.

Command Modes

Interface configuration

Command History

Release	Modification
12.2(15)T	This command was introduced.

Usage Guidelines

Use the **ip rsvp authentication type** command to specify the algorithm used to generate cryptographic signatures in RSVP messages. If you do not specify an algorithm, **md5** is used.

Examples

The following command sets the type to **sha-1**:

```
Router(config-if)# ip rsvp authentication type sha-1
```

Related Commands

Command	Description
ip rsvp authentication key	Specifies the key (string) for the RSVP authentication algorithm.

ip rsvp authentication window-size

To specify the maximum number of Resource Reservation Protocol (RSVP) authenticated messages that can be received out of order, use the **ip rsvp authentication window-size** command in interface configuration mode. To disable the window size (or to use the default value of 1), use the **no** form of this command.

ip rsvp authentication window-size [*n*]

no ip rsvp authentication window-size

Syntax Description	<i>n</i>	(Optional) Maximum number of authenticated messages that can be received out of order. The range is 1 to 64.
---------------------------	----------	--

Defaults The default value is 1.

Command Modes Interface configuration

Command History	Release	Modification
	12.2(15)T	This command was introduced.

Usage Guidelines Use the **ip rsvp authentication window-size** command to specify the maximum number of authenticated messages that can be received out of order. All RSVP authenticated messages include a sequence number that is used to prevent replays of RSVP messages.

With a default window size of one message, RSVP rejects any duplicate authenticated messages because they are assumed to be replay attacks. However, sometimes bursts of RSVP messages become reordered between RSVP neighbors. If this occurs on a regular basis, and you can verify that the node sending the burst of messages is trusted, you can use the **window-size** option to allow for the burst size such that RSVP will not discard such reordered bursts. RSVP will still check for duplicate messages within these bursts.

Examples The following command sets the window size to 2:

```
Router(config-if)# ip rsvp authentication window-size 2
```

Related Commands	Command	Description
	ip rsvp authentication	Activates RSVP cryptographic authentication.

ip rsvp bandwidth

To enable Resource Reservation Protocol (RSVP) for IP on an interface, use the **ip rsvp bandwidth** command in interface configuration mode. To disable RSVP completely, use the **no** form of this command. To eliminate only the subpool portion of the bandwidth, use the **no** form of this command with the keyword **sub-pool**.

ip rsvp bandwidth [*interface-kbps*] [*single-flow-kbps*] [**sub-pool** *kbps*]

no ip rsvp bandwidth [*interface-kbps*] [*single-flow-kbps*] [**sub-pool** *kbps*]

Syntax Description	
<i>interface-kbps</i>	(Optional) Maximum amount of bandwidth, in kbps, that may be allocated by RSVP flows. The range is from 1 to 10,000,000.
<i>single-flow-kbps</i>	(Optional) Maximum amount of bandwidth, in kbps, that may be allocated to a single flow. The range is from 1 to 10,000,000. This value is ignored by the Diff-Serv-aware MPLS Traffic Engineering feature available with Cisco IOS Release 12.2(4)T.
sub-pool <i>kbps</i>	(Optional) Amount of bandwidth in kbps on interface to be reserved to a portion of the total. The range is from 1 to the value of the <i>interface-kbps</i> argument.

Defaults

RSVP is disabled by default.

If the **ip rsvp bandwidth** command is entered but no bandwidth values are supplied (for example, **ip rsvp bandwidth** is entered followed by pressing the Enter key), a default bandwidth value (that is, 75% of the link bandwidth) is assumed for both the *interface-kbps* and *single-flow-kbps* arguments.

Command Modes

Interface configuration

Command History

Release	Modification
11.2	This command was introduced.
12.0(11)ST	The sub-pool keyword was added.
12.2(4)T	This command was integrated into Cisco IOS Release 12.2(4)T. This command was implemented on the Cisco 7500 series and the ATM-permanent virtual circuit (PVC) interface.

Usage Guidelines

RSVP cannot be configured with distributed Cisco Express Forwarding (dCEF).

RSVP is disabled by default to allow backward compatibility with systems that do not implement RSVP.

Weighted Random Early Detection (WRED) or fair queuing must be enabled first.

Examples

The following example shows a T1 (1536 kbps) link configured to permit RSVP reservation of up to 1158 kbps, but no more than 100 kbps for any given flow on serial interface 0. Fair queueing is configured with 15 reservable queues to support those reserved flows, should they be required.

```
Router(config)# interface serial 0
Router(config-if)# fair-queue 64 256 15
Router(config-if)# ip rsvp bandwidth 1158 100
```

Related Commands

Command	Description
fair-queue (WFQ)	Enables WFQ for an interface.
ip rsvp neighbor	Enables neighbors to request a reservation.
ip rsvp reservation	Enables a router to behave like it is receiving and forwarding RSVP RESV messages.
ip rsvp sender	Enables a router to behave like it is receiving and forwarding RSVP PATH messages.
ip rsvp udp-multicasts	Instructs the router to generate UDP-encapsulated RSVP multicasts whenever it generates an IP-encapsulated multicast packet.
random-detect (interface)	Enables WRED or DWRED.
show ip rsvp installed	Displays RSVP-related installed filters and corresponding bandwidth information.
show ip rsvp interface	Displays RSVP-related interface information.
show ip rsvp neighbor	Displays current RSVP neighbors.
show ip rsvp reservation	Displays RSVP-related receiver information currently in the database.
show ip rsvp sender	Displays RSVP PATH-related sender information currently in the database.

ip rsvp burst policing

To configure a burst factor within the Resource Reservation Protocol (RSVP) token bucket policer on a per-interface basis, use the **ip rsvp burst policing** command in interface configuration mode. To return to the default value, enter the **no** form of this command.

ip rsvp burst policing [*factor*]

no ip rsvp burst policing

Syntax Description	<i>factor</i>	(Optional) Indicates a burst factor value as a percentage of the requested burst of the receiver.
---------------------------	---------------	---

Defaults	The default value is 200; the minimum value is 100, and the maximum value is 700.
-----------------	---

Command Modes	Interface configuration
----------------------	-------------------------

Command History	Release	Modification
	12.1(3)T	This command was introduced.

Usage Guidelines	<p>You configure the burst police factor per interface, not per flow. The burst factor controls how strictly or loosely the traffic of the sender is policed with respect to burst.</p> <p>The burst factor applies to all RSVP flows installed on a specific interface. You can configure each interface independently for burst policing.</p>
-------------------------	---

Examples	<p>Here is an example of the ip rsvp burst policing command with a burst factor of 200:</p> <pre>ip rsvp burst policing 200</pre>
-----------------	--

ip rsvp data-packet classification none

To turn off (disable) Resource Reservation Protocol (RSVP) data packet classification, use the **ip rsvp data-packet classification none** command in interface configuration mode. To turn on (enable) data-packet classification, use the **no** form of this command.

ip rsvp data-packet classification none

no ip rsvp data-packet classification

Syntax Description This command has no arguments or keywords.

Defaults Disabled

Command Modes Interface configuration

Release	Modification
12.2(2)T	This command was introduced.

Usage Guidelines Use the **ip rsvp data-packet classification none** command when you do not want RSVP to process every packet. Configuring RSVP so that not every packet is processed eliminates overhead and improves network performance and scalability.

Examples This section contains two examples of the **ip rsvp data-packet classification none** command. In the first example, data packet classification is turned off (disabled), as follows:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# int atm6/0
Router(config-if)# ip rsvp data-packet classification none
```

In the second example, data packet classification is turned on (enabled), as follows:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# int atm6/0
Router(config-if)# no ip rsvp data-packet classification
```

Command	Description
show ip rsvp interface	Displays RSVP-related interface information.

ip rsvp dsbm candidate

To configure an interface as a Designated Subnetwork Bandwidth Manager (DSBM) candidate, use the **ip rsvp dsbm candidate** command in interface configuration mode. To disable DSBM on an interface, which exempts the interface as a DSBM candidate, use the **no** form of this command.

ip rsvp dsbm candidate [*priority*]

no ip rsvp dsbm candidate

Syntax Description

<i>priority</i>	(Optional) A value in the range from 64 to 128. Among contenders for the DSBM, the interface with the highest priority number wins the DSBM election process.
-----------------	---

Defaults

An interface is not configured as a DSBM contender by default. If you use this command to enable the interface as a DSBM candidate and you do not specify a priority, the default priority of 64 is assumed.

Command Modes

Interface configuration

Command History

Release	Modification
12.0(5)T	This command was introduced.
12.1(1)T	This command was integrated into Cisco IOS Release 12.1(1)T.

Usage Guidelines

SBM protocol entities, any one of which can manage resources on a segment, can reside in Layer 2 or Layer 3 devices. Many SBM-capable devices may be attached to a shared Layer 2 segment. When more than one SBM exists on a given segment, one of the SBMs is elected to be the DSBM. The elected DSBM is responsible for exercising admission control over requests for resource reservations on a segment, which, in the process, becomes a managed segment. A managed segment includes those interconnected parts of a shared LAN that are not separated by DSBMs. In all circumstances, only one, if any, DSBM exists for each Layer 2 segment.

You can configure an interface to have a DSBM priority in the range from 64 to 128. You can exempt an interface from participation in the DSBM election on a segment but still allow the system to interact with the DSBM if a DSBM is present on the segment. In other words, you can allow a Resource Reservation Protocol (RSVP)-enabled interface on a router connected to a managed segment to be managed by the DSBM even if you do not configure that interface to participate as a candidate in the DSBM election process. To exempt an interface from DSBM candidacy, do not issue the **ip rsvp dsbm candidate** command on that interface.

RSVP cannot be configured with VIP-distributed Cisco Express Forwarding (dCEF).

Examples

The following example configures Ethernet interface 2 as a DSBM candidate with a priority of 100:

```
interface Ethernet2
 ip rsvp dsbm candidate 100
```

Related Commands	Command	Description
	debug ip rsvp	Displays information about SBM message processing, the DSBM election process, and standard RSVP enabled message processing information.
	debug ip rsvp detail	Displays detailed information about RSVP and SBM.
	debug ip rsvp detail sbm	Displays detailed information about SBM messages only, and SBM and DSBM state transitions.
	ip rsvp dsbm non-resv-send-limit	Configures the NonResvSendLimit object parameters.
	show ip rsvp sbm	Displays information about an SBM configured for a specific RSVP-enabled interface or for all RSVP-enabled interfaces on the router.

ip rsvp dsbm non-resv-send-limit

To configure the NonResvSendLimit object parameters, use the **ip rsvp dsbm non-resv-send-limit** command in interface configuration mode. To use the default NonResvSendLimit object parameters, use the **no** form of this command.

```
ip rsvp dsbm non-resv-send-limit { rate kbps | burst kilobytes | peak kbps | min-unit bytes | max-unit bytes }
```

```
no ip rsvp dsbm non-resv-send-limit { rate kbps | burst kilobytes | peak kbps | min-unit bytes | max-unit bytes }
```

Syntax Description

rate <i>kbps</i>	The average rate, in kbps, for the Designated Subnetwork Bandwidth Manager (DSBM) candidate. The average rate is a number from 1 to 2147483.
burst <i>kilobytes</i>	The maximum burst size, in kb, for the DSBM candidate. The maximum burst size is a number from 1 to 2147483.
peak <i>kbps</i>	The peak rate, in kbps, for the DSBM candidate. The peak rate is a number from 1 to 2147483.
min-unit <i>bytes</i>	The minimum policed unit, in bytes, for the DSBM candidate. The minimum policed unit is a number from 1 to 2147483647.
max-unit <i>bytes</i>	The maximum packet size, in bytes, for the DSBM candidate. The maximum packet size is a number from 1 to 2147483647.

Defaults

The default for the **rate**, **burst**, **peak**, **min-unit**, and **max-unit** keywords is unlimited; all traffic can be sent without a valid Resource Reservation Protocol (RSVP) reservation.

Command Modes

Interface configuration

Command History

Release	Modification
12.1(1)T	This command was introduced.

Usage Guidelines

To configure the per-flow limit on the amount of traffic that can be sent without a valid RSVP reservation, configure the **rate**, **burst**, **peak**, **min-unit**, and **max-unit** values for finite values greater than 0.

To allow all traffic to be sent without a valid RSVP reservation, configure the **rate**, **burst**, **peak**, **min-unit**, and **max-unit** values for unlimited traffic. To configure the parameters for unlimited traffic, you can either omit the command, or enter the **no** form of the command (for example, **no ip rsvp dsbm non-resv-send-limit rate**). Unlimited is the default value.

The absence of the NonResvSendLimit object allows any amount of traffic to be sent without a valid RSVP reservation.

RSVP cannot be configured with VIP-distributed Cisco Express Forwarding (dCEF).

Examples

The following example configures Ethernet interface 2 as a DSBM candidate with a priority of 100, an average rate of 500 kbps, a maximum burst size of 1000 KB, a peak rate of 500 kbps, and unlimited minimum and maximum packet sizes:

```
interface Ethernet2
 ip rsvp dsbm candidate 100
 ip rsvp dsbm non-resv-send-limit rate 500
 ip rsvp dsbm non-resv-send-limit burst 1000
 ip rsvp dsbm non-resv-send-limit peak 500
```

Related Commands

Command	Description
ip rsvp dsbm candidate	Configures an interface as a DSBM candidate.
show ip rsvp sbm	Displays information about an SBM configured for a specific RSVP-enabled interface or for all RSVP-enabled interfaces on the router.

ip rsvp flow-assist

To enable Resource Reservation Protocol (RSVP) to attach itself to NetFlow so that it can leverage NetFlow services to obtain flow classification information about packets in order to update its token bucket and set IP Precedence as required, use the **ip rsvp flow-assist** command in interface configuration mode. To detach RSVP from NetFlow, use the **no** form of this command.

ip rsvp flow-assist

no ip rsvp flow-assist

Syntax Description This command has no arguments or keywords.

Defaults This command has no default behavior or values. (RSVP does not use NetFlow as a packet filtering mechanism.)

Command Modes Interface configuration

Command History	Release	Modification
	12.0(3)T	This command was introduced.

Usage Guidelines For RSVP to maintain token buckets and set IP Precedence on packets traversing the flow, it must interact with the underlying packet forwarding mechanism in order to obtain the information it needs. RSVP uses NetFlow for this purpose.

If RSVP is used on non-ATM links and RSVP must set IP Precedence without relying on traffic policing, weighted fair queueing (WFQ) cannot be used. In this case, a method of attaching RSVP to the underlying forwarding mechanism is required. The **ip rsvp flow-assist** command satisfies this requirement. It allows RSVP to attach itself to NetFlow so that it can use NetFlow to obtain information about packets, which it can then use to update its token bucket and set IP Precedence. NetFlow does not police packets or flows. For this reason, when RSVP is configured in this mode, it can only set IP Precedence and not otherwise police traffic.

In summary, you should use this command only when all of the following conditions exist:

- You want to set IP Precedence and type of service (ToS) bits using the **ip rsvp precedence** command or the **ip rsvp tos** command.
- You are not running WFQ on the interface.
- You are not running ATM or you have not specified the **ip rsvp svc-required** command.

When all of these conditions prevail, RSVP is completely detached from the data flow path and, thus, has no way to detect packets. Use of this command enables RSVP to detect packets so that it can mark them.

RSVP cannot be configured with VIP-distributed Cisco Express Forwarding (dCEF).

Use the **show ip rsvp interface** command to determine whether this command is in effect for an interface or subinterface.

Examples

The following example enables RSVP on the ATM interface 2/0/0 to attach itself to NetFlow:

```
interface atm2/0/0
 ip rsvp flow-assist
```

Related Commands

Command	Description
ip rsvp precedence	Allows you to set the IP Precedence values to be applied to packets that either conform to or exceed the RSVP flowspec.
ip rsvp svc-required	Enables creation of an SVC to service any new RSVP reservation made on the interface or subinterface.
ip rsvp tos	Allows you to set the ToS values to be applied to packets that either conform to or exceed the RSVP flowspec.
show ip rsvp interface	Displays RSVP-related interface information.

ip rsvp layer2 overhead

To control the overhead accounting performed by Resource Reservation Protocol (RSVP)/weighted fair queueing (WFQ) when a flow is admitted onto an ATM permanent virtual circuit (PVC), use the **ip rsvp layer2 overhead** command in interface configuration mode. To disable the overhead accounting, use the **no** form of this command.

ip rsvp layer2 overhead [*h c n*]

no ip rsvp layer2 overhead [*h c n*]

Syntax Description		
	<i>h</i>	(Optional) Layer 2 encapsulation header plus trailer size applied to each Layer 3 packet in bytes. Valid sizes are numbers from 0 to 65535.
	<i>c</i>	(Optional) Layer 2 cell header size applied to each Layer 2 cell in bytes. Valid sizes are numbers from 0 to 65535.
	<i>n</i>	(Optional) Layer 2 payload size in bytes. Valid sizes are numbers from 0 to 65534.

Defaults

This command is enabled by default on ATM interfaces that are running RSVP and WFQ. You can also use this command on non-ATM interfaces.

The default version of the command, which you specify by entering the default prefix, **default ip rsvp layer2 overhead**, or by omitting the parameters (*h*, *c*, and *n*) and entering the **ip rsvp layer2 overhead** command causes RSVP to determine the overhead values automatically, based on the interface/PVC encapsulation. (Currently, RSVP recognizes ATM Adaptation Layer 5 (AAL5) subnetwork access protocol (SNAP) and MUX (multiplexer) encapsulations.)

On non-ATM/PVC interfaces, the configured *h*, *c*, and *n* parameters determine the values that RSVP uses for its overhead.

Command Modes

Interface configuration

Command History

Release	Modification
12.2(2)T	This command was introduced.

Usage Guidelines

When an IP flow traverses a link, the overhead of Layer 2 encapsulation can increase the amount of bandwidth that the flow requires to exceed the advertised (Layer 3) rate.

In many cases, the additional bandwidth a flow requires because of Layer 2 overhead is negligible and can be transmitted as part of the 25 percent of the link, which is unreservable and kept for routing updates and Layer 2 overhead. This situation typically occurs when the IP flow uses large packet sizes or when the Layer 2 encapsulation allows for frames of variable size (such as in Ethernet and Frame Relay encapsulations).

However, when a flow's packet sizes are small and the underlying Layer 2 encapsulation uses fixed-size frames, the Layer 2 encapsulation overhead can be significant, as is the case when Voice Over IP (VoIP) flows traverse ATM links.

To avoid oversubscribing ATM PVCs, which use AAL5 SNAP or AAL5 MUX encapsulations, RSVP automatically accounts for the Layer 2 overhead when admitting a flow. For each flow, RSVP determines the total amount of bandwidth required, including Layer 2 overhead, and uses this value for admission control with the WFQ bandwidth manager.

**Note**

The **ip rsvp layer2 overhead** command does not affect bandwidth requirements of RSVP flows on ATM switched virtual circuits (SVCs).

Examples

In the following example, the total amount of bandwidth reserved with WFQ appears:

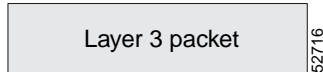
```
Router# show ip rsvp installed detail

RSVP:ATM6/0 has the following installed reservations
RSVP Reservation. Destination is 11.1.1.1, Source is 10.1.1.1,
  Protocol is UDP, Destination port is 1000, Source port is 1000
  Reserved bandwidth:50K bits/sec, Maximum burst:1K bytes, Peak rate:50K bits/sec
  Min Policed Unit:60 bytes, Max Pkt Size:60 bytes
  Resource provider for this flow:
    WFQ on ATM PVC 100/101 on AT6/0: PRIORITY queue 40. Weight:0, BW 89 kbps
  Conversation supports 1 reservations
  Data given reserved service:0 packets (0M bytes)
  Data given best-effort service:0 packets (0 bytes)
  Reserved traffic classified for 9 seconds
  Long-term average bitrate (bits/sec):0M reserved, 0M best-effort
```

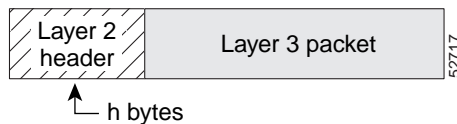
In the preceding example, the flow's advertised Layer 3 rate is 50 kbps. This value is used for admission control with the **ip rsvp bandwidth** value. The actual bandwidth required, inclusive of Layer 2 overhead, is 89 kbps. WFQ uses this value for admission control.

Typically, you should not need to configure or disable the Layer 2 overhead accounting. RSVP uses the advertised Layer 3 flow rate, minimum packet size, and maximum unit size in conjunction with the Layer 2 encapsulation characteristics of the ATM PVC to compute the required bandwidth for admission control. However, you can disable or customize the Layer 2 overhead accounting (for any link type) with the **ip rsvp layer2 overhead** command. The parameters of this command are based on the following steps that show how a Layer 3 packet is fragmented and encapsulated for Layer 2 transmission:

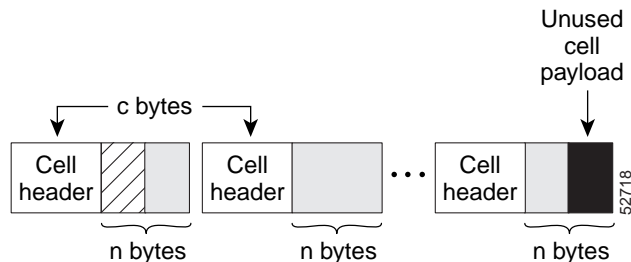
Step 1 Start with a Layer 3 packet, as shown in [Figure 1](#), which includes an IP header and a payload.

Figure 1 Layer 3 Packet

Step 2 Add an encapsulation header or trailer, as shown in [Figure 2](#), of size h .

Figure 2 Layer 3 Packet with Layer 2 Header

Step 3 Segment the resulting packet into fixed-sized cells, as shown in [Figure 3](#), with a cell header of c bytes and a cell payload of n bytes.

Figure 3 Segmented Packet

Step 4 Transmit the resulting Layer 2 cells.

More Configuration Examples

In the following example, Layer 2 overhead accounting is disabled for all reservations on the interface and its PVCs:

```
Router(config-if)# no ip rsvp layer2 overhead
```

In the following example, Layer 2 overhead accounting is configured with ATM AAL5 SNAP encapsulation:

```
Router(config-if)# no ip rsvp layer2 overhead 8 5 48
```

In the following example, Layer 2 overhead accounting is configured with ATM AAL5 MUX encapsulation:

```
Router(config-if)# ip rsvp layer2 overhead 0 5 48
```

In the following example, Layer 2 overhead accounting is configured with Ethernet V2.0 encapsulation (including 8-byte preamble, 6-byte source-active (SA) messages, 6-byte destination-active (DA) messages, 2-byte type, and 4-byte frame check sequence (FCS) trailer):

```
Router(config-if)# ip rsvp layer2 overhead 26 0 1500
```

Related Commands

Command	Description
show ip rsvp installed	Displays RSVP-related installed filters and corresponding bandwidth information.

ip rsvp listener

To configure a Resource Reservation Protocol (RSVP) router to listen for Path messages, use the **ip rsvp listener** command in global configuration mode. To disable listening, use the **no** form of this command.

```
ip rsvp listener dst {udp | tcp | any | number} {any | dst-port} {announce | reply | reject}
```

```
no ip rsvp listener
```

Syntax Description	
<i>dst</i>	IP address of the receiving interface.
udp tcp any <i>number</i>	User Datagram Protocol (UDP), TCP or any protocol to be used on the receiving interface and the UDP or TCP source port number. Note If you select <i>number</i> , the range is 0 to 255 and the protocol is IP.
any <i>dst-port</i>	Any destination port or a port number from 0 to 65535 for the receiving interface.
announce reply reject	Receiver announces the arrival of the flow at its destination, or sender requests a reply when flow is received, or router sends a PathError (reject) message in response to an incoming Path message that matches specified listener parameters.

Defaults	
	Disabled

Command Modes	
	Global configuration

Command History	Release	Modification
	12.2(13)T	This command was introduced.

Usage Guidelines

Use the **ip rsvp listener** command to find Path messages so that the router can proxy reservations. This command is similar to the **ip rsvp reservation** and **ip rsvp reservation-host** commands. However, they do not allow you to specify more than one port or protocol per command so that you may have to enter many commands to proxy for a set of ports and protocols. In contrast, the **ip rsvp listener** command allows you to use a wildcard for a set of ports and protocols by using just that one command.

Examples

In the following example, the sender is requesting that the receiver reply with a Resv message for the flow:

```
Router# configure terminal
```

```
Enter configuration commands, one per line. End with CNTL/Z.
```

```
Router(config)# ip rsvp listener 192.168.2.1 any any reply
```

Related Commands	Command	Description
	ip rsvp reservation	Enables a router to simulate receiving and forwarding RSVP Resv messages.
	ip rsvp reservation-host	Enables a router to simulate a host generating RSVP Resv messages.
	show ip rsvp listeners	Displays configured RSVP listeners.

ip rsvp neighbor

To enable neighbors to request a reservation, use the **ip rsvp neighbor** command in interface configuration mode. To disable this function, use the **no** form of this command.

ip rsvp neighbor *access-list-number*

no ip rsvp neighbor *access-list-number*

Syntax Description	<i>access-list-number</i>	Number of a standard or extended IP access list. It can be any number in the range from 1 to 199.
Defaults	The router accepts messages from any neighbor.	
Command Modes	Interface configuration	
Command History	Release	Modification
	11.2	This command was introduced.
Usage Guidelines	<p>Use this command to allow only specific Resource Reservation Protocol (RSVP) neighbors to make a reservation. If no limits are specified, any neighbor can request a reservation. If an access list is specified, only neighbors meeting the specified access list requirements can make a reservation.</p> <p>RSVP cannot be configured with VIP-distributed Cisco Express Forwarding (dCEF).</p>	
Examples	<p>The following example allows neighbors meeting access list 1 requirements to request a reservation:</p> <pre>interface ethernet 0 ip rsvp neighbor 1</pre>	
Related Commands	Command	Description
	fair-queue (WFQ)	Enables WFQ for an interface.
	ip rsvp bandwidth	Enables RSVP for IP on an interface.
	ip rsvp reservation	Enables a router to simulate receiving and forwarding RSVP RESV messages.
	ip rsvp sender	Enables a router to simulate receiving and forwarding RSVP PATH messages.
	ip rsvp udp-multicasts	Instructs the router to generate UDP-encapsulated RSVP multicasts whenever it generates an IP-encapsulated multicast packet.
	random-detect (interface)	Enables WRED or DWRED.

Command	Description
show ip rsvp installed	Displays RSVP-related installed filters and corresponding bandwidth information.
show ip rsvp interface	Displays RSVP-related interface information.
show ip rsvp neighbor	Displays current RSVP neighbors.
show ip rsvp reservation	Displays RSVP-related receiver information currently in the database.
show ip rsvp sender	Displays RSVP PATH-related sender information currently in the database.

ip rsvp policy cops minimal

To lower the load of the Common Open Policy Service (COPS) server and to improve latency times for messages on the governed router, use the **ip rsvp policy cops minimal** command in global configuration mode to restrict the COPS RSVP policy to adjudicate only PATH and RESV messages. To turn off the restriction, use the **no** form of this command.

ip rsvp policy cops minimal

no ip rsvp policy cops minimal

Syntax Description This command has no arguments or keywords.

Defaults The default state is OFF, causing all adjudicable RSVP messages to be processed by the configured COPS policy.

Command Modes Global configuration

Command History	Release	Modification
	12.1(1)T	This command was introduced.

Usage Guidelines When this command is used, COPS does not attempt to adjudicate PATHERROR and RESVERROR messages. Instead, those messages are all accepted and forwarded.

Examples In the following example, COPS authentication is restricted to PATH and RESV messages:

```
ip rsvp policy cops minimal
```

In the following example, that restriction is removed:

```
no ip rsvp policy cops minimal
```

ip rsvp policy cops report-all

To enable a router to report on its success and failure with outsourcing decisions, use the **ip rsvp policy cops report-all** command in global configuration mode. To return the router to its default, use the **no** form of this command.

ip rsvp policy cops report-all

no ip rsvp policy cops report-all

Syntax Description

This command has no arguments or keywords.

Defaults

The default state of this command is to send reports to the Policy Decision Point (PDP) about configuration decisions only.

Command Modes

Global configuration

Command History

Release	Modification
12.1(1)T	This command was introduced.

Usage Guidelines

In the default state, the router reports to the PDP when the router has succeeded or failed to implement Resource Reservation Protocol (RSVP) configuration decisions.

A *configuration decision* contains at least one of the following:

- A RESV ALLOC context (with or without additional contexts)
- A stateless or named decision object

A decision that does not contain at least one of those elements is an *outsourcing decision*.

Some brands of policy server might expect reports about RSVP messaging, which the default state of the Cisco Common Open Policy Service (COPS) for RSVP does not issue. In such cases, use the **ip rsvp policy cops report-all** command to ensure interoperability between the router and the policy server. Doing so does not adversely affect policy processing on the router.

Unicast FF reservation requests always stimulate a report from the router to the PDP, because those requests contain a RESV ALLOC context (combined with an IN CONTEXT and an OUT CONTEXT).

Examples

In order to show the Policy Enforcement Point (PEP)-to-PDP reporting process, the **debug cops** command in the following example already is enabled when a new PATH message arrives at the router:

```
Router(config)# ip rsvp policy cops report-all

Router(config)# 00:02:48:COPS:** SENDING MESSAGE **
Contents of router's request to PDP:
  COPS HEADER:Version 1, Flags 0, Opcode 1 (REQ), Client-type:1, Length:216
  HANDLE (1/1) object. Length:8.    00 00 02 01
  CONTEXT (2/1) object. Length:8.    R-type:5.    M-type:1
  IN_IF (3/1) object. Length:12.    Address:10.1.2.1.    If_index:4
  OUT_IF (4/1) object. Length:12.    Address:10.33.0.1.    If_index:3
  CLIENT SI (9/1) object. Length:168.    CSI data:
  [A 27-line Path message omitted here]
00:02:48:COPS:Sent 216 bytes on socket,
00:02:48:COPS:Message event!
00:02:48:COPS:State of TCP is 4
00:02:48:In read function
00:02:48:COPS:Read block of 96 bytes, num=104 (len=104)
00:02:48:COPS:** RECEIVED MESSAGE **
Contents of PDP's decision received by router:
  COPS HEADER:Version 1, Flags 1, Opcode 2 (DEC), Client-type:1, Length:104
  HANDLE (1/1) object. Length:8.    00 00 02 01
  CONTEXT (2/1) object. Length:8.    R-type:1.    M-type:1
  DECISION (6/1) object. Length:8.    COMMAND cmd:1, flags:0
  DECISION (6/3) object. Length:56.    REPLACEMENT
  [A 52-byte replacement object omitted here]
  CONTEXT (2/1) object. Length:8.    R-type:4.    M-type:1
  DECISION (6/1) object. Length:8.    COMMAND cmd:1, flags:0
00:02:48:Notifying client (callback code 2)
00:02:48:COPS:** SENDING MESSAGE **
Contents of router's report to PDP:
  COPS HEADER:Version 1, Flags 1, Opcode 3 (RPT), Client-type:1, Length:24
  HANDLE (1/1) object. Length:8.    00 00 02 01
  REPORT (12/1) object. Length:8.    REPORT type COMMIT (1)
00:02:48:COPS:Sent 24 bytes on socket,
```

ip rsvp policy cops servers

To specify that Resource Reservation Protocol (RSVP) should use Common Open Policy Service (COPS) policy for remote adjudication, use the **ip rsvp policy cops servers** command in global configuration mode. To turn off the use of COPS for RSVP, use the **no** form of this command.

```
ip rsvp policy cops [acl] servers server-ip [server-ip]
```

```
no ip rsvp policy cops [acl] servers
```

Syntax Description

<i>acl</i>	(Optional) Specifies the access control list (ACL) whose sessions will be governed by the COPS policy.
<i>server-ip</i>	Specifies the IP addresses of the servers governing the COPS policy. As many as eight servers can be specified, with the first being treated as the primary server.

Defaults

If no ACL is specified, the default behavior is for all reservations to be governed by the specified policy servers.

Command Modes

Global configuration

Command History

Release	Modification
12.1(1)T	This command was introduced.

Usage Guidelines

If more than one server is specified, the first server is treated by RSVP as the primary server, and functions as such for *all* ACLs specified.

All servers in the list must have the same policy configuration.

If the connection of the router to the server breaks, the router tries to reconnect to that same server. If the reconnection attempt fails, the router then obeys the following algorithm:

If the connection to the Policy Decision Point (PDP) is closed (either because the PDP closed the connection, a TCP/IP error occurred, or the keepalives failed), the Policy Enforcement Point (PEP) issues a CLIENT-CLOSE message and then attempts to reconnect to the same PDP. If the PEP receives a CLIENT-CLOSE message containing a PDP redirect address, the PEP attempts to connect to the redirected PDP. Note the following points:

- If either attempt fails, the PEP attempts to connect to the PDPs previously specified in the **ip rsvp policy cops servers** configuration command, obeying the sequence of servers given in that command, always starting with the first server in that list.
- If the PEP reaches the end of the list of servers without connecting, it waits a certain time (called the *reconnect delay*) before trying again to connect to the first server in the list. This reconnect delay is initially 30 seconds, and doubles each time the PEP reaches the end of the list without having connected, until the reconnect delay becomes its maximum of 30 minutes. As soon as a connection is made, the delay is reset to 30 seconds.

The **no** form of this command need not contain any server IP addresses, but it must contain *all* the previously specified access lists (see the last example in the following section).

Examples

This first example applies the COPS policy residing on server 172.27.224.117 to all reservations passing through router-9. It also identifies the backup COPS server for this router as the one at address 172.27.229.130:

```
Router(config)# ip rsvp policy cops servers 172.27.224.117 172.27.229.130
```

The next example applies the COPS policy residing on server 172.27.224.117 to reservations passing through router-9 only if they match access lists 40 and 160. Other reservations passing through that router will not be governed by this server. The command statement also identifies the backup COPS server for that router to be the one at address 172.27.229.130:

```
Router(config)# ip rsvp policy cops 40 160 servers 172.27.224.117 172.27.229.130
```

The following example turns off COPS for the previously specified access lists 40 and 160 (you cannot turn off just one of the previously specified lists):

```
Router(config)# no ip rsvp policy cops 40 160 servers
```

ip rsvp policy cops timeout

To configure the amount of time the Policy Enforcement Point (PEP) router will retain policy information after losing connection with the Common Open Policy Service (COPS) server, use the **ip rsvp policy cops timeout** command in global configuration mode. To restore the router to the default value (5 minutes), use the **no** form of this command.

ip rsvp policy cops timeout *policy-timeout*

no ip rsvp policy cops timeout

Syntax Description	<i>policy-timeout</i>	Duration of timeout, from 1 to 10,000 seconds.
--------------------	-----------------------	--

Defaults	Timeout default is 300 seconds (5 minutes).
----------	---

Command Modes	Global configuration
---------------	----------------------

Command History	Release	Modification
	12.1(1)T	This command was introduced.

Examples	The following example configures the router to time out all policy information relating to a lost server in 10 minutes:
----------	---

```
ip rsvp policy cops timeout 600
```

The following example resets the timeout to the default value:

```
no ip rsvp policy cops timeout
```

ip rsvp policy default-reject

To reject all messages that do not match the policy access control lists (ACLs), use the **ip rsvp policy default-reject** command in global configuration mode. To restore the default behavior, which passes along all messages that do not match the ACLs, use the **no** form of this command.

ip rsvp policy default-reject

no ip rsvp policy default-reject

Syntax Description

This command has no arguments or keywords.

Defaults

Without this command, the default behavior of Resource Reservation Protocol (RSVP) is to accept, install, or forward all unmatched RSVP messages. Once this command is invoked, all unmatched RSVP messages are rejected.

Command Modes

Global configuration

Command History

Release	Modification
12.1(1)T	This command was introduced.

Usage Guidelines

If COPS is configured without an ACL, or if any policy ACL is configured to use the **permit ip any any** command, the behavior of that ACL will take precedence, and no session will go unmatched.



Note

This command makes one exception to its blocking of unmatched messages. It forwards RESVERROR and PATHERROR messages that were generated by its own rejection of RESV and PATH messages. That is done to ensure that the default-reject operation does not remain totally hidden from network managers.



Caution

Be extremely careful with this command. It will shut down *all* RSVP processing on the router if access lists are too narrow or if no Common Open Policy Service (COPS) server has been specified. (Use the **ip rsvp policy cops servers** command to specify a COPS server.)

Examples

The following example configures RSVP to reject all unmatched reservations:

```
ip rsvp policy default-reject
```

The following example configures RSVP to accept all unmatched reservations:

```
no ip rsvp policy default-reject
```

ip rsvp policy local

To create a local procedure that determines the use of Resource Reservation Protocol (RSVP) resources in a network, use the **ip rsvp policy local** command in global configuration mode. To disable this function, use the **no** form of this command.

```
ip rsvp policy local { default | acl acl [acl1...acl8] }
```

```
no ip rsvp policy local
```

Syntax Description	default	Used when an RSVP message does not match any access control list (ACL).
	acl <i>acl</i> [<i>acl1...acl8</i>]	Used when an ACL is specified. Values for each IP ACL are from 1 to 199.
	Note	You must associate at least one ACL with an ACL-based policy. However, you can associate as many as eight.

Defaults	Disabled
----------	----------

Command Modes	Global configuration
---------------	----------------------

Command History	Release	Modification
	12.2(13)T	This command was introduced.

Usage Guidelines

Use the **ip rsvp policy local** command to create a local procedure that determines the use of RSVP resources in a network.

There are two types of local policies—one default local policy and one or more ACL-based local policies. The default policy is used when an RSVP message does not match any ACL-based policies. You can use local policies in the following combinations:

- A default policy and no ACL-based policies. All RSVP messages, regardless of reservation (data flow) source or destination, are subject to whatever is defined in this one policy.
- ACL-based policies and no default policy. If an RSVP message does not match the ACLs of any of these local policies, RSVP sees if there are any remote policies in place that allow the router to pass the RSVP message to a Common Open Policy Service (COPS) server for an accept/reject decision. If there are no COPS servers, the RSVP message is accepted. This final decision can be changed to a reject decision with the **ip rsvp policy default-reject** command.
- A default policy and ACL-based policies. If an RSVP message does not match the ACLs of any of these local policies, RSVP will carry out whatever decisions are in the default local policy.

An ACL-based policy must have at least one ACL associated with it, but it can optionally have up to eight ACLs. The ACLs can be standard or extended IP ACLs. They are matched against source/destination addresses/ports based on RSVP objects inside RSVP signaling messages, not on the IP headers of the RSVP messages.

CLI Submodes

After you type the **ip rsvp policy local default** or the **ip rsvp policy local acl** command, you enter local policy CLI submode where you define the properties of the default or ACL-based local policy that you are creating.



Note

The local policy that you create automatically rejects all RSVP messages unless you enter a submode command that instructs RSVP on the types of messages to accept.

The submode commands are as follows:

accept—Accepts, but does not forward RSVP messages.

accept { all | path | path-error | resv | resv-error }

- **all**—Accepts all RSVP messages.
- **path**—Accepts incoming Path messages that match the ACL(s) of this policy. If you omit this command, incoming Path messages that match the ACL(s) are rejected and a PathError message is sent in reply. However, the PathError reply is also subject to local policy.
- **path-error**—Accepts incoming PathError messages that match the ACL(s) of this policy. If you omit this command, incoming PathError messages that match the ACL(s) are rejected.
- **resv**—Accepts incoming Resv messages that match the ACL(s) of this policy and performs any required admission control. If you omit this command, incoming Resv messages that match the ACL(s) are rejected and a ResvError message is sent in reply. However, the ResvError reply is also subject to local policy.
- **resv-error**—Accepts incoming ResvError messages that match the ACL(s) of this policy. If you omit this command, the incoming ResvError messages matching the ACL(s) are rejected.
- **default**—Sets a command to its defaults.
- **exit**—Exits local policy configuration mode.
- **forward**—Accepts and forwards RSVP messages.

forward { all | path | path-error | resv | resv-error }

- **all**—Accepts and forwards all RSVP messages.
- **path**—Accepts and forwards Path messages that match the ACL(s) of this policy. If you omit this command, Path messages matching the ACL(s) are not forwarded to the next (downstream) hop.
- **path-error**—Accepts and forwards PathError messages that match the ACL(s) of this policy. If you omit this command, the PathError message matching the ACL(s) are not forwarded to the previous (upstream) hop. You may want to reject outbound PathError messages if you are receiving Path messages from an untrusted node because someone could be trying to port-scan for RSVP. If you reply with a PathError message, then the untrusted node knows you support RSVP and your IP address. Such information could be used to attempt RSVP-based attacks.
- **resv**—Accepts and forwards Resv messages that match the ACL(s) of this policy. If you omit this command, Resv messages matching the ACL(s) are not forwarded to the previous (upstream) hop.
- **resv-error**—Accepts and forwards ResvError messages that match the ACL(s) of this policy. If you omit this command, the ResvError message matching the ACL(s) is not forwarded to the next (downstream) hop. You may want to reject outbound ResvError messages if you are receiving Resv messages from an untrusted node because it could be someone trying to port-scan for RSVP. If you reply with a ResvError message, then the untrusted node knows you support RSVP and your IP address. Such information could be used to attempt RSVP-based attacks.

- **local-override**—Overrides any remote (COPS) policy by enforcing the local policy in effect. Finalizes any decisions by this policy. If local-override is omitted, RSVP holds on to the local policy decision to see if a remote (COPS) policy exists that will make a decision on the RSVP message, and only if there is no remote policy decision will the local policy decision be enforced.
- **no**—Negates a command or sets its defaults.
- **preempt-priority** <start-priority> [<hold-priority>]—Indicates the priorities for resource requests contained in Resv messages that match the ACL(s) of this policy. The range of priority values is 0 to 65,535.

The *start-priority* argument indicates the priority of the reservation when it is initially installed. The *hold-priority* argument indicates the priority of the reservation after it has been installed. When the *start-priority* argument is higher than the *hold-priority* argument, new reservations can steal bandwidth from longer-lived reservations; however, the start and hold priorities are often configured to be the same value. In order for reservations to be preempted in favor of reservations with higher priorities, there must be no RSVP bandwidth remaining on the interface the Resv message was received on, and a global **ip rsvp policy preempt** command must be issued. RSVP will preempt the first so many lower-priority reservations whose combined bandwidth meets (or exceeds) the amount of bandwidth required by a new, incoming, higher-priority reservation.

Label switched path (LSP) sessions are ignored when you select reservations to be preempted, because LSP sessions have their own preemption priority scheme that is configured with the **tunnel mpls traffic-eng priority** command.

In non-LSP sessions, RSVP reservations that are installed on a particular interface are searched in the following order to determine if they are eligible for preemption at a specific preemption priority:

- Destination address
- IP protocol type
- Destination port
- Source address (fixed-filter (FF) style reservations only)
- Source port (FF style reservations only)
- Downstream hop address (for shared media only; for example, Ethernet)

The above fields are searched from lower to higher values. The source address and source port fields are not checked for shared-explicit (SE) or wildcard-filter (WF) style reservations.



Note

If you exit local policy submode without entering any submode commands, the policy you have created will reject *all* RSVP messages.

Examples

In the following example, any RSVP nodes in the 192.168.101.0 subnet can initiate or respond to reservation requests, but all other nodes can respond only to reservation requests. This means that any 192.168.101.x node can send and receive Path, PathError, Resv, or ResvError messages. All other nodes can send only Resv or ResvError messages.

```
Router# configure terminal
```

Enter configuration commands, one per line. End with CNTL/Z.

```
Router(config)# access-list 104 permit ip 192.168.101.0 0.0.0.255 any
Router(config)# ip rsvp policy local acl 104
Router(config-rsvp-policy-local)# forward all
Router(config-rsvp-policy-local)# exit
```

■ ip rsvp policy local

```

Router(config)# ip rsvp policy local default
Router(config-rsvp-policy-local)# forward resv
Router(config-rsvp-policy-local)# forward resverror
Router(config-rsvp-policy-local)# end

```

Related Commands

Command	Description
ip rsvp policy preempt	Enables RSVP to take bandwidth from lower-priority reservations and give it to new, higher-priority reservations.
show ip rsvp policy	Displays the configured local policies.
show ip rsvp policy cops	Displays the policy server address(es), ACL IDs, and current state of the router server connection.
show ip rsvp policy local	Displays selected local policies that have been configured.
tunnel mpls traffic-eng priority	Configures the setup and reservation priority for an MPLS Traffic Engineering tunnel.

ip rsvp policy preempt

To enable Resource Reservation Protocol (RSVP) to take bandwidth from lower-priority reservations and give it to new, higher-priority reservations, use the **ip rsvp policy preempt** command in global configuration mode. To disable this function, use the **no** form of this command.

ip rsvp policy preempt

no ip rsvp policy preempt

Syntax Description This command has no arguments or keywords.

Defaults Disabled

Command Modes Global configuration

Release	Modification
12.2(13)T	This command was introduced.

Usage Guidelines Use the **ip rsvp policy preempt** command to enable or disable the preemption parameter for all configured local and remote policies without setting the preemption parameter for each policy individually. This command allows you to give preferential quality of service (QoS) treatment to one group of RSVP hosts or applications over another.

Examples The following example enables preemption:

```
Router(config)# ip rsvp policy preempt
```

The following example disables preemption:

```
Router(config)# no ip rsvp policy preempt
```

Command	Description
show ip rsvp policy	Displays the configured local policies.

ip rsvp pq-profile

To specify the criteria for Resource Reservation Protocol (RSVP) to use to determine which flows to direct into the priority queue (PQ) within weighted fair queueing (WFQ), use the **ip rsvp pq-profile** command in global configuration mode. To disable the specified criteria, use the **no** form of this command.

ip rsvp pq-profile [*voice-like* | *r'* [*b'* [*p-to-r'* | *ignore-peak-value*]]]

no ip rsvp pq-profile

Syntax Description		
<i>voice-like</i>	(Optional)	Indicates pq-profile parameters sufficient for most voice flows. The default values for <i>r'</i> , <i>b'</i> , and <i>p-to-r'</i> are used. These values should cause all voice flows generated from Cisco IOS applications and most voice flows from other RSVP applications, such as Microsoft NetMeeting, to be directed into the PQ.
<i>r'</i>	(Optional)	Indicates maximum rate of a flow in bytes per second. Valid range is from 1 to 1048576 bytes per second.
<i>b'</i>	(Optional)	Indicates maximum burst of a flow in bytes. Valid range is from 1 to 8192 bytes.
<i>p-to-r'</i>	(Optional)	Indicates maximum ratio of peak rate to average rate as a percentage. Valid range is from 100 to 4000 percent.
<i>ignore-peak-value</i>	(Optional)	Indicates that the peak rate to average rate ratio of the flow is not evaluated when RSVP identifies flows.

Defaults

The default value for *r'* is 12288 bytes per second.

The default value for *b'* is 592 bytes.

The default value for *p-to-r'* is 110 percent.

Command Modes

Global configuration

Command History

Release	Modification
12.1(3)T	This command was introduced.

Usage Guidelines

Use this command to define the profile of RSVP flows to be placed in the PQ within the WFQ system. You can have only one profile in effect at a time. Changes to this configuration affect only new flows, not existing flows.

This command applies only on interfaces that are running RSVP and WFQ.

RSVP recognizes voice flows based upon the r, b, and p values within the flowspec of a receiver. A reserved flow is granted to the PQ as long as the flowspec parameters of a receiver meet the following default criteria:

$(r \leq r') \text{ AND } (b \leq b') \text{ AND } (p/r \leq p\text{-to-}r')$

Examples

In the following example, voice-like flows (with the default criteria for voice) are put into the PQ:

```
Router(config)# ip rsvp pq-profile
Router(config)# ip rsvp pq-profile voice-like
Router(config)# ip rsvp pq-profile 12288 592 110
Router(config)# default ip rsvp pq-profile
Router# show run | include pq-profile
```

In the following example, all flows matching the voice criteria are put into the PQ:

```
Router(config)# ip rsvp pq-profile 10240 512 100
Router# show run | include pq-profile
```

```
ip rsvp pq-profile 10240 512 100
```

In the following example, no flows are put into the PQ:

```
Router(config)# no ip rsvp pq-profile
Router# show run | include pq-profile
```

```
no ip rsvp pq-profile
```

In the following example, flows with the criteria given for r' and b' and the default value for p-to-r' are put into the PQ:

```
Router(config)# ip rsvp pq-profile 9000 300
Router# show run | include pq-profile
```

```
ip rsvp pq-profile 9000 300 110
```

In the following example, flows with the criteria given for r' and b' and ignoring the peak value of the flow are put into the PQ:

```
Router(config)# ip rsvp pq-profile 9000 300 ignore-peak-value
Router# show run | include pq-profile
ip rsvp pq-profile 9000 300 ignore-peak-value
```

In the following example, Microsoft NetMeeting voice flows with G.711 or adaptive differential pulse code modulation (ADPCM) codecs are put into the PQ:

```
Router(config)# ip rsvp pq-profile 10200 1200
```

ip rsvp precedence

To enable the router to mark the IP Precedence value of the type of service (ToS) byte for packets in a Resource Reservation Protocol (RSVP) reserved path using the specified values for packets that either conform to or exceed the RSVP flowspec, use the **ip rsvp precedence** command in interface configuration mode. To remove existing IP Precedence settings, use the **no** form of this command.

ip rsvp precedence {[**conform** *precedence-value*] [**exceed** *precedence-value*]}

no ip rsvp precedence [**conform**] [**exceed**]

Syntax Description

conform *precedence-value* (Optional) Specifies an IP Precedence value in the range from 0 to 7 for traffic that conforms to the RSVP flowspec. The IP Precedence value is written to the three high-order bits (bits 5 to 7) of the ToS byte in the IP header of a packet. Either the **conform** or **exceed** keyword is required; both keywords may be specified.

When used with the **no** form of the command, the **conform** keyword is optional.

exceed *precedence-value* (Optional) Specifies an IP Precedence value in the range from 0 to 7 for traffic that exceeds the RSVP flowspec. The IP Precedence value is written to the three high-order bits (bits 5 to 7) of the ToS byte in the IP header of a packet. Either the **conform** or **exceed** keyword is required; both keywords may be specified.

When used with the **no** form of the command, the **exceed** keyword is optional.

Defaults

The IP Precedence bits of the ToS byte are left unmodified when this command is not used. The default state is equivalent to execution of the **no ip rsvp precedence** command.

If neither the **conform** nor **exceed** keyword is specified, all IP Precedence settings are removed.

Command Modes

Interface configuration

Command History

Release	Modification
12.0(3)T	This command was introduced.

Usage Guidelines

Packets in an RSVP reserved path are divided into two classes: those that conform to the reservation flowspec and those that correspond to a reservation but that exceed, or are outside, the reservation flowspec.

The **ip rsvp precedence** command allows you to set the IP Precedence values to be applied to packets belonging to these two classes. You must specify the IP Precedence value for at least one class of traffic when you use this command. You can use a single instance of the command to specify values for both classes, in which case you can specify the **conform** and **exceed** keywords in either order.

As part of its input processing, RSVP uses the **ip rsvp precedence** command to set the IP Precedence bits on conforming and nonconforming packets. If per-VC DWRED is configured, the system uses the IP Precedence and ToS bit settings on the output interface in its packet drop process. The IP Precedence setting of a packet can also be used by interfaces on downstream routers.

Execution of the **ip rsvp precedence** command causes IP Precedence values for all preexisting reservations on the interface to be modified.

**Note**

RSVP must be enabled on an interface before you can use this command; that is, use of the **ip rsvp bandwidth** command must precede use of the **ip rsvp precedence** command. RSVP cannot be configured with VIP-distributed Cisco Express Forwarding (dCEF).

RSVP receives packets from the underlying forwarding mechanism. Therefore, before you use the **ip rsvp precedence** command to set IP Precedence, one of the following features is required:

- Weighted fair queueing (WFQ) must be enabled on the interface.
- RSVP switched virtual circuits (SVCs) must be used.
- NetFlow must be configured to assist RSVP.

**Note**

Use of the **no** form of this command is not equivalent to giving the **ip rsvp precedence 0** command, which sets all precedence on the packets to 0, regardless of previous precedence setting.

Examples

The following example sets the IP Precedence value to 3 for all traffic on the ATM interface 0 that conforms to the RSVP flowspec and to 2 for all traffic that exceeds the flowspec:

```
interface atm0
 ip rsvp precedence conform 3 exceed 2
```

The following example sets the IP Precedence value to 2 for all traffic on ATM interface 1 that conforms to the RSVP flowspec. The IP Precedence values of those packets that exceed the flowspec are not altered in any way.

```
interface ATM1
 ip rsvp precedence conform 2
```

Related Commands

Command	Description
ip rsvp bandwidth	Enables RSVP for IP on an interface.
ip rsvp policy cops minimal	Lowens the COPS server's load and improves latency times for messages on the governed router.
ip rsvp tos	Allows you to set the ToS values to be applied to packets that either conform to or exceed the RSVP flowspec.
show ip rsvp	Displays the IP Precedence and ToS bit values to be applied to packets that either conform to or exceed the RSVP flowspec for a given interface.

ip rsvp reservation

To enable a router to simulate receiving and forwarding Resource Reservation Protocol (RSVP) RESV messages, use the **ip rsvp reservation** command in global configuration mode. To disable this function, use the **no** form of this command.

```
ip rsvp reservation session-ip-address sender-ip-address {tcp | udp | ip-protocol} session-dport
sender-sport next-hop-ip-address next-hop-interface {ff | se | wf} {rate | load} bandwidth
burst-size
```

```
no ip rsvp reservation session-ip-address sender-ip-address {tcp | udp | ip-protocol}
session-dport sender-sport next-hop-ip-address next-hop-interface {ff | se | wf} {rate | load}
bandwidth burst-size
```

Syntax Description

<i>session-ip-address</i>	For unicast sessions, this is the address of the intended receiver; for multicast sessions, this is the IP multicast address of the session.
<i>sender-ip-address</i>	The IP address of the sender.
tcp udp <i>ip-protocol</i>	TCP, User Datagram Protocol (UDP), or IP protocol in the range from 0 to 255.
<i>session-dport</i> <i>sender-sport</i>	<i>session-dport</i> is the destination port. <i>sender-sport</i> is the source port. Port numbers are specified in all cases, because the use of 16-bit ports following the IP header is not limited to UDP or TCP. If destination is zero, source must be zero, and the implication is that ports are not checked. If destination is nonzero, source must be nonzero (except for wf reservations, for which the source port is always ignored and can therefore be zero).
<i>next-hop-ip-address</i>	Host name or address of the receiver or the router closest to the receiver.
<i>next-hop-interface</i>	Next hop interface or subinterface type and number. Interface type can be ethernet , loopback , null , or serial .
ff se wf	Reservation style: <ul style="list-style-type: none"> Fixed Filter (ff) is single reservation. Shared Explicit (se) is shared reservation, limited scope. Wild Card Filter (wf) is shared reservation, unlimited scope.
rate load	QoS guaranteed bit rate service or controlled load service.
<i>bandwidth</i>	Average bit rate, in kbps, to reserve up to 75 percent of the total on the interface. The range is from 1 to 10000000.
<i>burst-size</i>	Maximum burst size (KB of data in queue). The range is from 1 to 65535.

Defaults

The router does not simulate receiving and processing RSVP RESV messages by default.

Command Modes

Global configuration

Command History

Release	Modification
11.2	This command was introduced.

Usage Guidelines

Use this command to make the router simulate receiving RSVP RESV messages from a downstream host. This command can be used to proxy RSVP RESV messages for non-RSVP-capable receivers. By giving a local (loopback) next hop address and next hop interface, you can also use this command to proxy RSVP for the router you are configuring.

**Note**

RSVP cannot be configured with VIP-distributed Cisco Express Forwarding (dCEF).

Examples

The following example specifies the use of a Shared Explicit style of reservation and the controlled load service, with token buckets of 100 or 150 kbps and 60 or 65 kbps maximum queue depth:

```
ip rsvp reservation 224.250.0.2 172.16.1.1 UDP 20 30 172.16.4.1 Et1 se load 100 60
ip rsvp reservation 224.250.0.2 172.16.2.1 TCP 20 30 172.16.4.1 Et1 se load 150 65
```

The following example specifies the use of a Wild Card Filter style of reservation and the guaranteed bit rate service, with token buckets of 300 or 350 kbps and 60 or 65 kbps maximum queue depth:

```
ip rsvp reservation 224.250.0.3 0.0.0.0 UDP 20 0 172.16.4.1 Et1 wf rate 300 60
ip rsvp reservation 226.0.0.1 0.0.0.0 UDP 20 0 172.16.4.1 Et1 wf rate 350 65
```

Note that the Wild Card Filter does not admit the specification of the sender; it accepts all senders. This action is denoted by setting the source address and port to zero. If, in any filter style, the destination port is specified to be zero, RSVP does not permit the source port to be anything else; it understands that such protocols do not use ports or that the specification applies to all ports.

Related Commands

Command	Description
fair-queue (WFQ)	Enables WFQ for an interface.
ip rsvp bandwidth	Enables RSVP for IP on an interface.
ip rsvp neighbor	Enables neighbors to request a reservation.
ip rsvp reservation-host	Enables a router to simulate a host generating RSVP RESV messages.
ip rsvp sender	Enables a router to simulate receiving and forwarding RSVP PATH messages.
ip rsvp sender-host	Enables a router to simulate a host generating RSVP PATH messages.
ip rsvp udp-multicasts	Instructs the router to generate UDP-encapsulated RSVP multicasts whenever it generates an IP-encapsulated multicast packet.
random-detect (interface)	Enables WRED or DWRED.
show ip rsvp installed	Displays RSVP-related installed filters and corresponding bandwidth information.
show ip rsvp interface	Displays RSVP-related interface information.
show ip rsvp neighbor	Displays current RSVP neighbors.
show ip rsvp reservation	Displays RSVP-related receiver information currently in the database.
show ip rsvp sender	Displays RSVP PATH-related sender information currently in the database.

ip rsvp reservation-host

To enable a router to simulate a host generating Resource Reservation Protocol (RSVP) RESV messages, use the **ip rsvp reservation-host** command in global configuration mode. To disable this function, use the **no** form of this command.

```
ip rsvp reservation-host session-ip-address sender-ip-address {tcp | udp | ip-protocol}
session-dport sender-sport {ff | se | wf} {rate | load} bandwidth burst-size
```

```
no ip rsvp reservation-host session-ip-address sender-ip-address {tcp | udp | ip-protocol}
session-dport sender-sport {ff | se | wf} {rate | load} bandwidth burst-size
```

Syntax Description		
<i>session-ip-address</i>		For unicast sessions, this is the address of the intended receiver. IP multicast addresses cannot be used with this argument. It must be a logical address configured on an interface on the router you are configuring.
<i>sender-ip-address</i>		The IP address of the sender.
tcp udp <i>ip-protocol</i>		TCP, User Datagram Protocol UDP, or IP protocol in the range from 0 to 255.
<i>session-dport</i> <i>sender-sport</i>		<i>session-dport</i> is the destination port. <i>sender-sport</i> is the source port. Port numbers are specified in all cases, because the use of 16-bit ports following the IP header is not limited to UDP or TCP. If destination is zero, source must be zero, and the implication is that ports are not checked. If destination is nonzero, source must be nonzero (except for wf reservations, for which the source port is always ignored and can therefore be zero).
ff se wf		Reservation style: <ul style="list-style-type: none"> Fixed Filter (ff) is single reservation. Shared Explicit (se) is shared reservation, limited scope. Wild Card Filter (wf) is shared reservation, unlimited scope.
rate load		QoS guaranteed bit rate service or controlled load service.
<i>bandwidth</i>		Average bit rate, in kbps, to reserve up to 75 percent of the total on the interface. The range is from 1 to 10000000.
<i>burst-size</i>		Maximum burst size (KB of data in queue). The range is from 1 to 65535.

Defaults The router does not simulate a host generating RSVP RESV messages by default.

Command Modes Global configuration

Command History	Release	Modification
	12.0	This command was introduced.

Usage Guidelines

Use this command to make the router simulate a host generating its own RSVP RESV messages. This command is similar to the **ip rsvp reservation** command, which can cause the router to generate RESV messages on behalf of another host.

The main differences between the **ip rsvp reservation-host** and **ip rsvp reservation** commands follow:

- When you enter the **ip rsvp reservation-host** command, the *session-ip-address* argument must be a local address configured on an interface on the router. Therefore, you cannot proxy a reservation on behalf of a flow destined for another host. Also, you cannot use this command to generate reservation messages for multicast sessions.
- Because the message is assumed to originate from the router you are configuring, you do not specify a next hop or incoming interface for the RSVP RESV message when entering the **ip rsvp reservation-host** command.

Because you cannot use the command to proxy RSVP for non-RSVP-capable hosts or for multicast sessions, the **ip rsvp reservation-host** command is used mostly for debugging and testing purposes.

RSVP cannot be configured with VIP-distributed Cisco Express Forwarding (dCEF).

Examples

The following example specifies the use of a Shared Explicit style of reservation and the controlled load service, with token buckets of 100 or 150 kbps and 60 or 65 kbps maximum queue depth:

```
ip rsvp reservation-host 10.1.1.1 10.30.1.4 UDP 20 30 se load 100 60
ip rsvp reservation-host 10.40.2.2 10.22.1.1 TCP 20 30 se load 150 65
```

Related Commands

Command	Description
fair-queue (WFQ)	Enables WFQ for an interface.
ip rsvp bandwidth	Enables RSVP for IP on an interface.
ip rsvp neighbor	Enables neighbors to request a reservation.
ip rsvp reservation	Enables a router to simulate receiving and forwarding RSVP RESV messages.
ip rsvp sender	Enables a router to simulate receiving and forwarding RSVP PATH messages.
ip rsvp sender-host	Enables a router to simulate a host generating RSVP PATH messages.
ip rsvp udp-multicasts	Instructs the router to generate UDP-encapsulated RSVP multicasts whenever it generates an IP-encapsulated multicast packet.
random-detect (interface)	Enables WRED or DWRED.
show ip rsvp installed	Displays RSVP-related installed filters and corresponding bandwidth information.
show ip rsvp interface	Displays RSVP-related interface information.
show ip rsvp neighbor	Displays current RSVP neighbors.
show ip rsvp reservation	Displays RSVP-related receiver information currently in the database.
show ip rsvp sender	Displays RSVP PATH-related sender information currently in the database.

ip rsvp resource-provider

To configure a resource provider for an aggregate flow, use the **ip rsvp resource-provider** command in interface configuration mode. To disable a resource provider for an aggregate flow, use the **no** form of this command.

ip rsvp resource-provider { **none** | **wfq-interface** | **wfq-pvc** }

no ip rsvp resource-provider

Syntax Description	none	No resource provider specified regardless of whether one is configured on the interface.
	wfq-interface	Weighted fair queueing (WFQ) specified as the resource provider on the interface.
	wfq-pvc	WFQ specified as the resource provider on the permanent virtual circuit (PVC) or connection.

Defaults The **wfq interface** is the default resource provider that Resource Reservation Protocol (RSVP) configures on the interface.

Command Modes Interface configuration

Command History	Release	Modification
	12.2(2)T	This command was introduced.

Usage Guidelines Use the **ip rsvp resource-provider** command to configure the resource provider with which you want RSVP to interact when it installs a reservation.

To ensure that a flow receives quality of service (QoS) guarantees when using WFQ on a per-flow basis, configure **wfq-interface** or **wfq-pvc** as the resource provider. To ensure that a flow receives QoS guarantees when using class-based weighted fair queueing (CBWFQ) for data packet processing, configure **none** as the resource provider.



Note

Resource provider was formerly called QoS provider.

Examples In the following example, the **ip rsvp resource-provider** command is configured with **wfq-interface** or **wfq-pvc** as the resource provider, ensuring that a flow receives QoS guarantees when using WFQ on a per-flow basis:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# interface atm6/0
Router(config-if)# ip rsvp resource-provider wfq-pvc
```

In the following example, the **ip rsvp resource-provider** command is configured with **none** as the resource provider, ensuring that a flow receives QoS guarantees when using CBWFQ for data packet processing:

```
Router# configure terminal  
Enter configuration commands, one per line. End with CNTL/Z.  
Router(config)# interface atm6/0  
Router(config-if)# ip rsvp resource-provider none
```

Related Commands

Command	Description
show ip rsvp interface	Displays RSVP-related interface information.

ip rsvp sender

To enable a router to simulate receiving and forwarding Resource Reservation Protocol (RSVP) PATH messages, use the **ip rsvp sender** command in global configuration mode. To disable this function, use the **no** form of this command.

ip rsvp sender *session-ip-address sender-ip-address* {**tcp** | **udp** | *ip-protocol*} *session-dport sender-sport previous-hop-ip-address previous-hop-interface bandwidth burst-size*

no ip rsvp sender *session-ip-address sender-ip-address* {**tcp** | **udp** | *ip-protocol*} *session-dport sender-sport previous-hop-ip-address previous-hop-interface bandwidth burst-size*

Syntax Description		
<i>session-ip-address</i>		For unicast sessions, this is the address of the intended receiver; for multicast sessions, it is the IP multicast address of the session.
<i>sender-ip-address</i>		The IP address of the sender.
tcp udp <i>ip-protocol</i>		TCP, User Datagram Protocol (UDP), or IP protocol in the range from 0 to 255.
<i>session-dport</i> <i>sender-sport</i>		<i>session-dport</i> is the destination port. <i>sender-sport</i> is the source port. Port numbers are specified in all cases, because the use of 16-bit ports following the IP header is not limited to UDP or TCP. If destination is zero, source must be zero, and the implication is that ports are not checked. If destination is nonzero, source must be nonzero (except for wf reservations, for which the source port is always ignored and can therefore be zero).
<i>previous-hop-ip-address</i>		Address of the sender or the router closest to the sender.
<i>previous-hop-interface</i>		Address of the previous hop interface or subinterface. Interface type can be ethernet , loopback , null , or serial .
<i>bandwidth</i>		Average bit rate, in kbps, to reserve up to 75 percent of the total on the interface. The range is from 1 to 10000000.
<i>burst-size</i>		Maximum burst size (KB of data in queue). The range is from 1 to 65535.

Defaults The router does not simulate receiving and processing RSVP PATH messages by default.

Command Modes Global configuration

Command History	Release	Modification
	11.2	This command was introduced.

Usage Guidelines

Use this command to make the router simulate that it is receiving RSVP PATH messages from an upstream host. The command can be used to proxy RSVP PATH messages for non-RSVP-capable senders. By including a local (loopback) previous hop address and previous hop interface, you can also use this command to proxy RSVP for the router you are configuring.

RSVP cannot be configured with Versatile Interface Processor (VIP)-distributed Cisco Express Forwarding (dCEF).

Examples

The following example sets up the router to act like it is receiving RSVP PATH messages using UDP over loopback interface 1:

```
ip rsvp sender 224.250.0.1 172.16.2.1 udp 20 30 172.16.2.1 loopback 1 50 5
ip rsvp sender 224.250.0.2 172.16.2.1 udp 20 30 172.16.2.1 loopback 1 50 5
```

Related Commands

Command	Description
fair-queue (WFQ)	Enables WFQ for an interface.
ip rsvp bandwidth	Enables RSVP for IP on an interface.
ip rsvp neighbor	Enables neighbors to request a reservation.
ip rsvp reservation	Enables a router to simulate receiving and forwarding RSVP RESV messages.
ip rsvp reservation-host	Enables a router to simulate a host generating RSVP RESV messages.
ip rsvp sender-host	Enables a router to simulate a host generating RSVP PATH messages.
ip rsvp udp-multicasts	Instructs the router to generate UDP-encapsulated RSVP multicasts whenever it generates an IP-encapsulated multicast packet.
random-detect (interface)	Enables WRED or DWRED.
show ip rsvp installed	Displays RSVP-related installed filters and corresponding bandwidth information.
show ip rsvp interface	Displays RSVP-related interface information.
show ip rsvp neighbor	Displays current RSVP neighbors.
show ip rsvp reservation	Displays RSVP-related receiver information currently in the database.
show ip rsvp sender	Displays RSVP PATH-related sender information currently in the database.

ip rsvp sender-host

To enable a router to simulate a host generating a Resource Reservation Protocol (RSVP) PATH message, use the **ip rsvp sender-host** command in global configuration mode. To disable this function, use the **no** form of this command.

ip rsvp sender-host *session-ip-address sender-ip-address* { **tcp** | **udp** | *ip-protocol* } *session-dport sender-sport bandwidth burst-size*

no ip rsvp sender-host *session-ip-address sender-ip-address* { **tcp** | **udp** | *ip-protocol* } *session-dport sender-sport bandwidth burst-size*

Syntax Description

<i>session-ip-address</i>	For unicast sessions, this is the address of the intended receiver; for multicast sessions, it is the IP multicast address of the session.
<i>sender-ip-address</i>	The IP address of the sender. It must be a logical address configured on an interface on the router you are configuring.
tcp udp <i>ip-protocol</i>	TCP, User Datagram Protocol (UDP), or IP protocol in the range from 0 to 255.
<i>session-dport</i> <i>sender-sport</i>	<i>session-dport</i> is the destination port. <i>sender-sport</i> is the source port. Port numbers are specified in all cases, because the use of 16-bit ports following the IP header is not limited to UDP or TCP. If destination is zero, source must be zero, and the implication is that ports are not checked. If destination is nonzero, source must be nonzero (except for wf reservations, for which the source port is always ignored and can therefore be zero).
<i>bandwidth</i>	Average bit rate, in kbps, to reserve up to 75 percent of the total on the interface. The range is from 1 to 10000000.
<i>burst-size</i>	Maximum burst size (KB of data in queue). The range is from 1 to 65535.

Defaults

The router does not simulate RSVP PATH message generation by default.

Command Modes

Global configuration

Command History

Release	Modification
12.0	This command was introduced.

Usage Guidelines

Use this command to make the router simulate a host generating its own RSVP PATH messages. This command is similar to the **ip rsvp sender** command, which can cause the router to generate RSVP PATH messages on behalf of another host.

The main differences between the **ip rsvp sender-host** and **ip rsvp sender** commands follow:

- When you enter the **ip rsvp sender-host** command, the *sender-ip-address* argument must be a local address configured on an interface on the router.
- Because the message is assumed to originate from the router you are configuring, you do not specify a previous hop or incoming interface for the RSVP PATH message when entering the **ip rsvp sender-host** command.

Because you cannot use the command to proxy RSVP for non-RSVP-capable hosts, the **ip rsvp sender-host** command is used mostly for debugging and testing purposes.

RSVP cannot be configured with VIP-distributed Cisco Express Forwarding (dCEF).

Examples

The following example sets up the router to act like a host that will send traffic to the given multicast address:

```
ip rsvp sender-host 224.250.0.1 10.24.2.1 udp 20 30 50 5
ip rsvp sender-host 227.0.0.1 10.24.2.1 udp 20 30 50 5
```

Related Commands

Command	Description
fair-queue (WFQ)	Enables WFQ for an interface.
ip rsvp bandwidth	Enables RSVP for IP on an interface.
ip rsvp neighbor	Enables neighbors to request a reservation.
ip rsvp reservation	Enables a router to simulate receiving and forwarding RSVP RESV messages.
ip rsvp reservation-host	Enables a router to simulate a host generating RSVP RESV messages.
ip rsvp sender	Enables a router to simulate receiving and forwarding RSVP PATH messages.
ip rsvp udp-multicasts	Instructs the router to generate UDP-encapsulated RSVP multicasts whenever it generates an IP-encapsulated multicast packet.
random-detect (interface)	Enables WRED or DWRED.
show ip rsvp installed	Displays RSVP-related installed filters and corresponding bandwidth information.
show ip rsvp interface	Displays RSVP-related interface information.
show ip rsvp neighbor	Displays current RSVP neighbors.
show ip rsvp reservation	Displays RSVP-related receiver information currently in the database.
show ip rsvp sender	Displays RSVP PATH-related sender information currently in the database.

ip rsvp signalling dscp

To specify the differentiated services code point (DSCP) value to be used on all RSVP messages transmitted on an interface, use the **ip rsvp signalling dscp** command in interface configuration mode. To disable this function, use the **no** form of this command.

ip rsvp signalling dscp *value*

no ip rsvp signalling dscp

Syntax Description	<i>value</i>	Indicates a DSCP value. A DSCP value can be a number from 0 to 63.
---------------------------	--------------	--

Defaults	The default value is 0.
-----------------	-------------------------

Command Modes	Interface configuration
----------------------	-------------------------

Command History	Release	Modification
	12.1	This command was introduced

Usage Guidelines	<p>You configure the DSCP per interface, not per flow. The DSCP determines the priority that a packet receives from various hops as it travels to its destination.</p> <p>The DSCP applies to all RSVP flows installed on a specific interface. You can configure each interface independently for DSCP.</p>
-------------------------	--

Examples	Here is an example of the ip rsvp signalling dscp command with a DSCP value of 6:
-----------------	--

```
Router(config-if)# ip rsvp signalling dscp 6
Router# show ip rsvp interface detail serial2/0

Se2/0:
  Bandwidth:
    Curr allocated:10K bits/sec
    Max. allowed (total):1536K bits/sec
    Max. allowed (per flow):1536K bits/sec
  Neighbors:
    Using IP enacp:1. Using UDP encaps:0
    DSCP value used in Path/Resv msgs:0x6
    Burst Police Factor:300%
    RSVP:Data Packet Classification provided by: none
Router#
```

ip rsvp signalling initial-retransmit-delay

To configure the minimum amount of time that a Resource Reservation Protocol (RSVP)-configured router waits for an acknowledgment (ACK) message before retransmitting the same message, use the **ip rsvp signalling initial-retransmit-delay** command in global configuration mode. To reset the delay value to its default, use the **no** form of this command.

ip rsvp signalling initial-retransmit-delay *delay-value*

no ip rsvp signalling initial-retransmit-delay

Syntax Description	<i>delay-value</i>	Minimum amount of time that a router waits for an ACK message before the first retransmission of the same message. The delay value ranges from 500 to 30,000 milliseconds (ms).
---------------------------	--------------------	---

Defaults The default value is 1000 ms (1.0 sec).

Command Modes Global configuration

Command History	Release	Modification
	12.2(13)T	This command was introduced.

Usage Guidelines Use the **ip rsvp signalling initial-retransmit-delay** command to configure the minimum amount of time that a router waits for an ACK message before retransmitting the same message.

If an ACK is not received for a state, the first retransmit occurs after the initial retransmit interval. If no ACK is received after the first retransmit, a second retransmit occurs. The message continues to be retransmitted, with the gap between successive retransmits being twice the previous interval, until an ACK is received. Then the message drops into normal refresh schedule if it needs to be refreshed (Path and Resv messages), or is processed (Error or Tear messages). If no ACK is received after five retransmits, the message is discarded as required.

Examples The following command shows how to set the initial-retransmit-delay to 2 seconds:

```
Router(config)# ip rsvp signalling initial-retransmit-delay 2000
```

The following command shows how to reset the initial-retransmit-delay to the default (1.0 sec):

```
Router(config)# no ip rsvp signalling initial-retransmit-delay
```

ip rsvp signalling patherr state-removal

To reduce the amount of Resource Reservation Protocol (RSVP) traffic messages in a network, use the **ip rsvp signalling patherr state-removal** command in global configuration mode. To disable this function, use the **no** form of this command.

ip rsvp signalling patherr state-removal [**neighbor** *acl*]

no ip rsvp signalling patherr state-removal

Syntax Description		
	neighbor	(Optional) Adjacent routers that are part of a particular traffic engineering tunnel.
	<i>acl</i>	(Optional) A simple access list with values from 1 to 99.

Defaults Disabled

Command Modes Global configuration

Command History	Release	Modification
	12.2(13)T	This command was introduced.

Usage Guidelines Use the **ip rsvp signalling patherr state-removal** command to allow routers to delete Path state automatically when forwarding a PathError message, thereby eliminating the need for a subsequent PathTear message.

This command is most effective when all network nodes support this feature. All nodes need to have the latest version of Cisco IOS software configured.

This command applies only to label-switched path (LSP) flows.

Examples The following command shows how to enable **ip rsvp signalling patherr state-removal**:

```
Router(config)# ip rsvp signalling patherr state-removal
```

The following command shows how to disable **ip rsvp signalling patherr state-removal**:

```
Router(config)# no ip rsvp signalling patherr state-removal
```

The following command shows how to enable **ip rsvp signalling patherr state-removal** based on an access control list (ACL):

```
Router(config)# ip rsvp signalling patherr state-removal neighbor 98
```

The following command shows how to disable **ip rsvp signalling patherr state-removal** based on an ACL:

```
Router(config)# no ip rsvp signalling patherr state-removal neighbor 98
```

ip rsvp signalling rate-limit

To control the transmission rate for Resource Reservation Protocol (RSVP) messages sent to a neighboring router during a specified amount of time, use the **ip rsvp signalling rate-limit** command in global configuration mode. To disable this function, use the **no** form of this command.

ip rsvp signalling rate-limit [*burst*] [*maxsize*] [*period*]

no ip rsvp signalling rate-limit

Syntax Description		
<i>burst</i>	(Optional) Maximum number of RSVP messages allowed to be sent to a neighboring router during this interval. Range is 1 to 5000 messages. Default is 4 messages.	
<i>maxsize</i>	(Optional) Maximum size of the message queue in bytes. Range is 1 to 5000 bytes. Default is 500 bytes.	
<i>period</i>	(Optional) Length of the interval (timeframe) in milliseconds (ms). Range is 10 to 5000 ms. Default is 20 ms.	

Defaults Disabled

Command Modes Global configuration

Command History	Release	Modification
	12.2(13)T	This command was introduced.

Usage Guidelines Use the **ip rsvp signalling rate-limit** command to prevent a burst of RSVP traffic engineering signaling messages from overflowing the input queue of a receiving router, which would cause the router to drop some messages. Dropped messages substantially delay the completion of signaling.

Examples The following command shows how every 10 ms 6 messages with a message queue of 500 bytes are sent to any neighboring router:

```
Router(config)# ip rsvp signalling rate-limit 10 6 500
```

Related Commands	Command	Description
	debug ip rsvp rate-limit	Displays debug messages for RSVP rate-limiting events.

ip rsvp signalling refresh reduction

To enable Resource Reservation Protocol (RSVP) refresh reduction, use the **ip rsvp signalling refresh reduction** command in global configuration mode. To disable refresh reduction, use the **no** form of this command.

ip rsvp signalling refresh reduction

no ip rsvp signalling refresh reduction

Syntax Description This command has no arguments or keywords.

Defaults Disabled

Command Modes Global configuration

Command History	Release	Modification
	12.2(13)T	This command was introduced.

Usage Guidelines RSVP refresh reduction is a set of extensions to reduce the messaging load imposed by RSVP and to help it scale to support larger numbers of flows.

The following features of the refresh reduction standard (RFC 2961) are supported and will be turned on with this command:

- Setting the refresh-reduction-capable bit in message headers
- Message-Identifier (ID) usage
- Reliable messaging with rapid retransmit, acknowledgement (ACK) messages, and MESSAGE_ID objects
- Summary refresh extension
- Bundle messages (reception only)

Refresh reduction requires the cooperation of the neighbor to operate; for this purpose, the neighbor must also support the standard. If the router detects that a directly connected neighbor is not supporting the refresh reduction standard (either through observing the refresh-reduction-capable bit in messages received from the next hop, or by sending a MESSAGE_ID object to the next hop and receiving an error), refresh reduction will not be used on this link irrespective of this command.

Examples The following command shows how to enable RSVP refresh reduction:

```
Router(config)# ip rsvp signalling refresh reduction
```

The following command shows how to disable RSVP refresh reduction:

```
Router(config)# no ip rsvp signalling refresh reduction
```

Related Commands

Command	Description
show ip rsvp interface	Displays RSVP-related interface information.
show ip rsvp signalling refresh reduction	Displays refresh-reduction parameters for RSVP messages.

ip rsvp signalling refresh reduction ack-delay

To configure the maximum amount of time that a Resource Reservation Protocol (RSVP)-configured router holds on to an acknowledgment (ACK) message before sending it, use the **ip rsvp signalling refresh reduction ack-delay** command in global configuration mode. To reset the ack-delay value to its default, use the **no** form of this command.

ip rsvp signalling refresh reduction ack-delay *delay-value*

no ip rsvp signalling refresh reduction ack-delay

Syntax Description	<i>delay-value</i>	Maximum amount of time that a router holds on to an ACK message before sending it. Values range from 100 to 10000 milliseconds (ms).
---------------------------	--------------------	--

Defaults The default value is 250 ms (0.25 sec).

Command Modes Global configuration

Command History	Release	Modification
	12.2(13)T	This command was introduced.

Usage Guidelines Use the **ip rsvp signalling refresh reduction ack-delay** command to configure the maximum amount of time that an RSVP-configured router keeps an ACK message before sending it.

Examples The following command shows how to set the ack-delay value to 1 second:

```
Router(config)# ip rsvp signalling refresh reduction ack-delay 1000
```

The following command shows how to set the ack-delay value to the default value:

```
Router(config)# no ip rsvp signalling refresh reduction ack-delay
```

ip rsvp svc-required

To enable creation of a switched virtual circuit (SVC) to service any new Resource Reservation Protocol (RSVP) reservation made on the interface or subinterface of an Enhanced ATM port adapter (PA-A3), use the **ip rsvp svc-required** command in interface configuration mode. To disable SVC creation for RSVP reservations, use the **no** form of this command.

ip rsvp svc-required

no ip rsvp svc-required

Syntax Description This command has no arguments or keywords.

Defaults Disabled

Command Modes Interface configuration

Command History	Release	Modification
	12.0(3)T	This command was introduced.

Usage Guidelines This command applies exclusively to the RSVP-ATM QoS Interworking feature.

Usually reservations are serviced when RSVP classifies packets and a queueing mechanism schedules them for transmission to manage congestion. Traditionally, RSVP is used with weighted fair queueing (WFQ). When RSVP is coupled with WFQ, all of the packets visible to WFQ are also visible to RSVP, which allows RSVP to identify and take action on packets important to it. In this case, WFQ provides bandwidth guarantees.

However, when the **ip rsvp svc-required** command is used to configure an interface or subinterface, a new SVC is established and used to service each new reservation on the interface. ATM SVCs are used to provide bandwidth guarantees and NetFlow is used on input interfaces to make data packets visible to RSVP.



Note When RSVP is enabled, all packets are processed by the Route Switch Processor (RSP).

This command must be executed on both ends of an SVC driven by RSVP. This command is supported only for the Enhanced ATM port adapter (PA-A3) and its subinterfaces.



Note For this command to take effect, NetFlow must be enabled. Therefore, the **ip route-cache flow** command must precede this command in the configuration.

Use the **show ip rsvp interface** command to determine whether this command is in effect for any interface or subinterface.

ip rsvp svc-required

Examples

The following example signals RSVP that reservations made on ATM interface 2/0/0 will be serviced by creation of an SVC:

```
interface atm2/0/0
 ip rsvp svc-required
```

Related Commands

Command	Description
ip route-cache flow	Enables NetFlow switching for IP routing.
ip rsvp atm-peak-rate-limit	Sets a limit on the peak cell rate of reservations for all newly created RSVP SVCs established on the current interface or any of its subinterfaces.
ip rsvp precedence	Allows you to set the IP Precedence values to be applied to packets that either conform to or exceed the RSVP flowspec.
show ip rsvp interface	Displays RSVP-related interface information.

ip rsvp tos

To enable the router to mark the five low-order type of service (ToS) bits of the IP header ToS byte for packets in a Resource Reservation Protocol (RSVP) reserved path using the specified values for traffic that either conforms to or exceeds the RSVP flowspec, use the **ip rsvp tos** command in interface configuration mode. To remove existing settings for the ToS bits, use the **no** form of this command; if neither the **conform** nor **exceed** keyword is specified, all settings for the ToS bits are removed.

```
ip rsvp tos {[conform tos-value] [exceed tos-value]}
```

```
no ip rsvp tos [conform] [exceed]
```

Syntax Description

conform <i>tos-value</i>	(Optional) Specifies a ToS value in the range from 0 to 31 for traffic that conforms to the RSVP flowspec. The ToS value is written to the five low-order bits (bits 0 to 4) of the ToS byte in the IP header of a packet. Either the conform or exceed keyword is required; both keywords may be specified. When used with the no form of the command, the conform keyword is optional.
exceed <i>tos-value</i>	(Optional) Specifies a ToS value in the range from 0 to 31 for traffic that exceeds the RSVP flowspec. The ToS byte value is written to the five low-order bits (bits 0 to 4) of the ToS byte in the IP header of a packet. Either the conform or exceed keyword is required; both keywords may be specified. When used with the no form of the command, the exceed keyword is optional.

Defaults

The ToS bits of the ToS byte are left unmodified when this command is not used. (The default behavior is equivalent to use of the **no ip rsvp tos** command.)

Command Modes

Interface configuration

Command History

Release	Modification
12.0(3)T	This command was introduced.

Usage Guidelines

Packets in an RSVP reserved path are divided into two classes: those that conform to the reservation flowspec and those that correspond to a reservation but that exceed, or are outside, the reservation flowspec.

The **ip rsvp tos** command allows you to set the ToS values to be applied to packets belonging to these two classes. You must specify the ToS value for at least one class of traffic when you use this command. You can use a single instance of the command to specify values for both classes, in which case you can specify the **conform** and **exceed** keywords in either order.

As part of its input processing, RSVP uses the **ip rsvp tos** command configuration to set the ToS bits of the ToS byte on conforming and nonconforming packets. If per-virtual circuit (VC) VIP-distributed Weighted Random Early Detection (DWRED) is configured, the system uses the ToS bit and IP Precedence bit settings on the output interface in its packet drop process. The ToS bit and IP Precedence bit settings of a packet can also be used by interfaces on downstream routers.

Execution of the **ip rsvp tos** command causes ToS bit values for all preexisting reservations on the interface to be modified.

**Note**

RSVP must be enabled on an interface before you can use this command; that is, use of the **ip rsvp bandwidth** command must precede use of the **ip rsvp tos** command. RSVP cannot be configured with VIP-distributed Cisco Express Forwarding (dCEF).

**Note**

The **ip rsvp tos** command sets bits 0 to 4 so that in combination with the IP Precedence bit settings every bit in the ToS byte is set. Use of these bits is made with full knowledge of the fact that certain canonical texts that address the ToS byte specify that only bits 1 to 4 are used as the ToS bits.

RSVP receives packets from the underlying forwarding mechanism. Therefore, to use the **ip rsvp tos** command to set the ToS bits, one of the following features is required:

- Weighted fair queueing (WFQ) must be enabled on the interface.
- RSVP switched virtual circuits (SVCs) must be used.
- NetFlow must be configured to assist RSVP.

**Note**

Use of the **no** form of this command is not equivalent to giving the **ip rsvp tos 0** command, which sets all precedence on the packets to 0, regardless of previous precedence setting.

Examples

The following example sets the ToS bits value to 4 for all traffic on ATM interface 1 that conforms to the RSVP flowspec. ToS bits on packets exceeding the flowspec are not altered.

```
interface atm1
 ip rsvp tos conform 4
```

Related Commands

Command	Description
ip rsvp bandwidth	Enables RSVP for IP on an interface.
ip rsvp flow-assist	Enables RSVP to attach itself to NetFlow so that it can leverage NetFlow services.
ip rsvp policy cops minimal	Lowers the COPS server's load and improves latency times for messages on the governed router.
show ip rsvp	Displays the IP Precedence and ToS bit values to be applied to packets that either conform to or exceed the RSVP flowspec for a given interface.

ip rsvp udp-multicasts

To instruct the router to generate User Datagram Protocol (UDP)-encapsulated Resource Reservation Protocol (RSVP) multicasts whenever it generates an IP-encapsulated multicast packet, use the **ip rsvp udp-multicasts** command in interface configuration mode. To disable this function, use the **no** form of this command.

ip rsvp udp-multicasts [*multicast-address*]

no ip rsvp udp-multicasts [*multicast-address*]

Syntax Description	<i>multicast-address</i>	(Optional) Host name or UDP multicast address of router.
--------------------	--------------------------	--

Defaults	The generation of UDP multicasts is disabled. If a system sends a UDP-encapsulated RSVP message to the router, the router begins using UDP for contact with the neighboring system. The router uses multicast address 224.0.0.14 and starts sending to UDP port 1699. If the command is entered with no specifying multicast address, the router uses the same multicast address.
----------	---

Command Modes	Interface configuration
---------------	-------------------------

Command History	Release	Modification
	11.2	This command was introduced.

Usage Guidelines	Use this command to instruct a router to generate UDP-encapsulated RSVP multicasts whenever it generates an IP-encapsulated multicast packet. Some hosts require this trigger from the router. RSVP cannot be configured with VIP-distributed Cisco Express Forwarding (dCEF).
------------------	--

Examples	The following example reserves up to 7500 kbps on Ethernet interface 2, with up to 1 Mbps per flow. The router is configured to use UDP encapsulation with the multicast address 224.0.0.14.
----------	--

```
interface ethernet 2
 ip rsvp bandwidth 7500 1000
 ip rsvp udp-multicasts 224.0.0.14
```

Related Commands	Command	Description
	ip rsvp bandwidth	Enables RSVP for IP on an interface.
	ip rsvp neighbor	Enables neighbors to request a reservation.
	ip rsvp reservation	Enables a router to simulate receiving and forwarding RSVP RESV messages.
	ip rsvp sender	Enables a router to simulate receiving and forwarding RSVP PATH messages.

ip rtp compression-connections

To specify the total number of Real-Time Transport Protocol (RTP) header compression connections that can exist on an interface, use the **ip rtp compression-connections** command in interface configuration mode. To restore the default value, use the **no** form of this command.

ip rtp compression-connections *number*

no ip rtp compression-connections

Syntax Description

<i>number</i>	Number of RTP header compression connections the cache supports, in the range from 3 to 1000.
---------------	---

Defaults

For PPP and High-Level Data Link Control (HDLC) interfaces, the default is 16 compression connections.

For Frame Relay interfaces, the default is 256 compression connections.

Command Modes

Interface configuration

Command History

Release	Modification
11.3	This command was introduced.
12.0(7)T	For PPP and HDLC interfaces, the maximum number of compression connections increased from 256 to 1000. For Frame Relay interfaces, the maximum number of compression connections increased from 32 to 256. The default number of compression connections was increased from 32 (fixed) to 256 (configurable).

Usage Guidelines

You should configure one connection for each RTP call through the specified interface.

Each connection sets up a compression cache entry, so you are in effect specifying the maximum number of cache entries and the size of the cache. Too few cache entries for the specified interface can lead to degraded performance, and too many cache entries can lead to wasted memory.



Note

Both ends of the serial connection must use the same number of cache entries.

Examples

The following example changes the number of RTP header compression connections supported to 150:

```
Router> enable
Router# configure terminal
Router(config)# interface Serial1/0.0
Router(config-if)# encapsulation ppp
Router(config-if)# ip rtp header-compression
Router(config-if)# ip rtp compression-connections 150
Router(config-if)# exit
```

Related Commands

Command	Description
ip rtp header-compression	Enables RTP header compression.
show ip rtp header-compression	Displays RTP header compression statistics.

ip rtp header-compression

To enable Real-Time Transport Protocol (RTP) header compression, use the **ip rtp header-compression** command in interface configuration mode. To disable RTP header compression, use the **no** form of this command.

ip rtp header-compression [**passive** | **iphc-format** | **ietf-format**] [**periodic-refresh**]

no ip rtp header-compression [**passive** | **iphc-format** | **ietf-format**] [**periodic-refresh**]

Syntax Description

passive	(Optional) Compresses outgoing RTP packets only if incoming RTP packets on the same interface are compressed. If you do not specify the passive keyword, all RTP packets are compressed.
iphc-format	(Optional) Indicates that the IP Header Compression (IPHC) format of header compression will be used.
ietf-format	(Optional) Indicates that the Internet Engineering Task Force (IETF) format of header compression will be used.
periodic-refresh	(Optional) Indicates that the compressed IP header will be refreshed periodically.

Defaults

Disabled

For PPP interfaces, the default format for header compression is the IPHC format.

For High-Level Data Link Control (HDLC) and Frame Relay interfaces, the default format for header compression is the original proprietary Cisco format. The maximum number of compression connections for the proprietary Cisco format is 256.

Command Modes

Interface configuration

Command History

Release	Modification
11.3	This command was introduced.
12.0	This command was integrated into Cisco IOS Release 12.0. This command was modified to include the iphc-format keyword.
12.3(2)T	This command was integrated into Cisco IOS Release 12.3(2)T. This command was modified to include the periodic-refresh keyword.
12.3(4)T	This command was modified to include the ietf-format keyword.

Usage Guidelines

You can compress IP/User Datagram Protocol (UDP)/RTP headers to reduce the size of your packets. Compressing headers is especially useful for RTP because RTP payload size can be as small as 20 bytes, and the uncompressed header is 40 bytes.

Header Compression **passive** Keyword

By default, the **ip rtp header-compression** command compresses outgoing RTP traffic. This command includes an optional **passive** keyword. If you specify the **passive** keyword, outgoing RTP traffic is compressed only if *incoming* RTP traffic on the *same* interface is compressed. If you do not specify the **passive** keyword, *all* RTP traffic is compressed.

For PPP interfaces, the **passive** keyword is ignored. PPP interfaces negotiate the use of header-compression, regardless of whether the **passive** keyword is specified. Therefore, on PPP interfaces, the **passive** keyword is replaced by the IPHC format, the default format for PPP interfaces.

Header Compression **iphc-format** Keyword

This command includes the **iphc-format** keyword. The **iphc-format** keyword indicates the type of header compression that will be used. For PPP and HDLC interfaces, when the **iphc-format** keyword is specified, TCP header-compression is also enabled. For this reason, the **ip tcp header-compression** command appears in the output of the **show running-config** command. Since both RTP and TCP header compression are enabled, both UDP and TCP packets are compressed.

The **iphc-format** keyword includes checking whether the destination port number is even and in the ranges of 16385 to 32767 (for Cisco audio) or 49152 to 65535 (for Cisco video). Valid RTP packets that meet the criteria (that is, the port number is even and within the specified range) are compressed using the compressed RTP packet format. Otherwise, packets are compressed using the less-efficient compressed non-TCP packet format.



Note For Frame Relay interfaces, the **iphc-format** keyword is not available.

Header Compression **ietf-format** Keyword

This command includes the **ietf-format** keyword. The **ietf-format** keyword indicates the type of header compression that will be used. For HDLC interfaces, the **ietf-format** compresses only UDP packets. For PPP interfaces, when the **ietf-format** keyword is specified, TCP header-compression is also enabled. For this reason, the **ip tcp header-compression** command appears in the output of the **show running-config** command. Since both RTP and TCP header compression are enabled, both UDP and TCP packets are compressed.

However, with the **ietf-format** keyword, the requirement of checking whether a destination port number is in a specific range has been removed. Any even destination port number higher than 1024 can be used. Valid RTP packets that meet the criteria (that is, the port number is even and higher than 1024), are compressed using the compressed RTP packet format. Otherwise, packets are compressed using the less-efficient compressed non-TCP packet format.



Note For Frame Relay interfaces, the **ietf-format** keyword is not available.

Support for Serial Lines

RTP header compression is supported on serial lines using Frame Relay, HDLC, or PPP encapsulation. You must enable compression on both ends of a serial connection.

Unicast or Multicast RTP Packets

This command can compress unicast or multicast RTP packets, and, hence, multicast backbone (MBONE) traffic can also be compressed over slow links. The compression scheme is beneficial only when you have small payload sizes, as in audio traffic.

Examples

The following example enables RTP header compression on the Serial1/0.0 subinterface and limits the number of RTP header compression connections to 10. In this example, the optional **iphc-format** keyword of the **ip rtp header-compression** command is specified.

```
Router> enable
Router# configure terminal
Router(config)# interface Serial1/0.0
Router(config-if)# encapsulation ppp
Router(config-if)# ip rtp header-compression iphc-format
Router(config-if)# ip rtp compression-connections 10
Router(config-if)# exit
```

The following example enables RTP header compression on the Serial2/0.0 subinterface and limits the number of RTP header compression connections to 20. In this example, the optional **ietf-format** keyword of the **ip rtp header-compression** command is specified.

```
Router> enable
Router# configure terminal
Router(config)# interface Serial2/0.0
Router(config-if)# encapsulation ppp
Router(config-if)# ip rtp header-compression ietf-format
Router(config-if)# ip rtp compression-connections 20
Router(config-if)# exit
```

In the following example, RTP header compression is enabled on the Serial1/0.1 subinterface and the optional **periodic-refresh** keyword of the **ip rtp header-compression** command is specified:

```
Router> enable
Router# configure terminal
Router(config)# interface Serial1/0.1
Router(config-if)# encapsulation ppp
Router(config-if)# ip rtp header-compression iphc-format periodic-refresh
Router(config-if)# ip rtp compression-connections 10
Router(config-if)# exit
```

Related Commands

Command	Description
clear ip rtp header-compression	Clears RTP header compression structures and statistics.
ip rtp compression-connections	Specifies the total number of RTP header compression connections that can exist on an interface.
show ip rtp header-compression	Displays RTP header compression statistics.
show running-config	Displays the contents of the currently running configuration file or the configuration for a specific interface, or map class information.

ip tcp compression-connections

To specify the total number of Transmission Control Protocol (TCP) header compression connections that can exist on an interface, use the **ip tcp compression-connections** command in interface configuration mode. To restore the default, use the **no** form of this command.

ip tcp compression-connections *number*

no ip tcp compression-connections

Syntax Description	<i>number</i>	Number of TCP header compression connections the cache supports, in the range from 3 to 256.
---------------------------	---------------	--

Defaults	For PPP and High-Level Data Link Control (HDLC) interfaces, the default is 16 compression connections. For Frame Relay interfaces, the default is 256 compression connections.
-----------------	---

Command Modes	Interface configuration
----------------------	-------------------------

Command History	Release	Modification
	10.0	This command was introduced.
	12.0(7)T	For Frame Relay interfaces, the maximum number of compression connections increased from 32 to 256. The default number of compression connections was increased from 32 (fixed) to 256 (configurable).

Usage Guidelines	You should configure one connection for each TCP connection through the specified interface. Each connection sets up a compression cache entry, so you are in effect specifying the maximum number of cache entries and the size of the cache. Too few cache entries for the specified interface can lead to degraded performance, and too many cache entries can lead to wasted memory.
-------------------------	---



Note Both ends of the serial connection must use the same number of cache entries.

Examples	The following example sets the first serial interface for header compression with a maximum of ten cache entries:
-----------------	---

```
Router> enable
Router# configure terminal
Router(config)# interface serial 0
Router(config-if)# ip tcp header-compression
Router(config-if)# ip tcp compression-connections 10
Router(config-if)# exit
```

Related Commands	Command	Description
	ip tcp header-compression	Enables TCP header compression.
	show ip tcp header-compressions	Displays TCP header compression statistics.

ip tcp header-compression

To enable Transmission Control Protocol (TCP) header compression, use the **ip tcp header-compression** command in interface configuration mode. To disable compression, use the **no** form of this command.

ip tcp header-compression [**passive** | **iphc-format** | **ietf-format**]

no ip tcp header-compression [**passive** | **iphc-format** | **ietf-format**]

Syntax Description		
passive	(Optional) Compresses outgoing TCP packets only if incoming TCP packets on the same interface are compressed. If you do not specify the passive keyword, all TCP packets are compressed.	
iphc-format	(Optional) Indicates that the IP Header Compression (IPHC) format of header compression will be used.	
ietf-format	(Optional) Indicates that the Internet Engineering Task Force (IETF) format of header compression will be used.	

Defaults

Disabled

For PPP interfaces, the default format for header compression is the IPHC format.

For High-Level Data Link Control (HDLC) and Frame Relay interfaces, the default format is as described in RFC 1144, [Compressing TCP/IP Headers for Low-Speed Serial Links](#).

Command Modes

Interface configuration

Command History

Release	Modification
10.0	This command was introduced.
12.0	This command was integrated into Cisco IOS Release 12.0. This command was modified to include the iphc-format keyword.
12.3(4)T	This command was integrated into Cisco IOS Release 12.3(4)T. This command was modified to include the ietf-format keyword.

Usage Guidelines

You can compress the headers of your TCP/IP packets in order to reduce the size of your packets. TCP header compression is supported on serial lines using Frame Relay, HDLC, or PPP encapsulation. You must enable compression on both ends of a serial connection. Compressing the TCP header can speed up Telnet connections dramatically.

In general, TCP header compression is advantageous when your traffic consists of many small packets, not for traffic that consists of large packets. Transaction processing (usually using terminals) tends to use small packets and file transfers use large packets. This feature only compresses the TCP header, so it has no effect on User Datagram Protocol (UDP) packets or other protocol headers.

Header Compression **passive** Keyword

By default, the **ip tcp header-compression** command compresses outgoing TCP traffic. This command includes an optional **passive** keyword. If you specify the **passive** keyword, outgoing TCP traffic is compressed only if *incoming* TCP traffic on the *same* interface is compressed. If you do not specify the **passive** keyword, *all* TCP traffic is compressed.

For PPP interfaces, the **passive** keyword is ignored. PPP interfaces negotiate the use of header-compression, regardless of whether the **passive** keyword is specified. Therefore, on PPP interfaces, the **passive** keyword is replaced by the IPHC format, the default format for PPP interfaces.

Header Compression **iphc-format** Keyword

This command includes the **iphc-format** keyword. The **iphc-format** keyword indicates the type of header compression that will be used. For PPP and HDLC interfaces, when the **iphc-format** keyword is specified, RTP header-compression is also enabled. For this reason, the **ip rtp header-compression** command appears in the output of the **show running-config** command. Since both TCP and RTP header compression are enabled, both TCP and UDP packets are compressed.



Note For Frame Relay interfaces, the **iphc-format** keyword is not available.

Header Compression **ietf-format** Keyword

This command includes the **ietf-format** keyword. The **ietf-format** keyword indicates the type of header compression that will be used. For HDLC interfaces, the **ietf-format** compresses only TCP packets. For PPP interfaces, when the **ietf-format** keyword is specified, RTP header-compression is also enabled. For this reason, the **ip rtp header-compression** command appears in the output of the **show running-config** command. Since both TCP and RTP header compression are enabled, both TCP and UDP packets are compressed.



Note For Frame Relay interfaces, the **ietf-format** keyword is not available.

Examples

The following example sets the first serial interface for header compression with a maximum of ten cache entries:

```
Router> enable
Router# configure terminal
Router(config)# interface serial 0
Router(config-if)# ip tcp header-compression
Router(config-if)# ip tcp compression-connections 10
Router(config-if)# exit
```

The following example enables RTP header compression on the Serial1/0.0 subinterface and limits the number of RTP header compression connections to 10. In this example, the optional **iphc-format** keyword of the **ip tcp header-compression** command is specified.

```
Router> enable
Router# configure terminal
Router(config)# interface Serial1/0.0
Router(config-if)# encapsulation ppp
Router(config-if)# ip tcp header-compression iphc-format
Router(config-if)# ip tcp compression-connections 10
Router(config-if)# exit
```

The following example enables RTP header compression on the Serial2/0.0 subinterface and limits the number of RTP header compression connections to 20. In this example, the optional **ietf-format** keyword of the **ip tcp header-compression** command is specified.

```
Router> enable
Router# configure terminal
Router(config)# interface Serial2/0.0
Router(config-if)# encapsulation ppp
Router(config-if)# ip tcp header-compression ietf-format
Router(config-if)# ip tcp compression-connections 20
Router(config-if)# exit
```

Related Commands

Command	Description
ip tcp compression-connections	Specifies the total number of TCP header compression connections that can exist on an interface.
show ip tcp header-compression	Displays TCP header compression statistics.
show running-config	Displays the contents of the currently running configuration file or the configuration for a specific interface, or map class information.

ip rtp priority

To reserve a strict priority queue for a set of Real-Time Transport Protocol (RTP) packet flows belonging to a range of User Datagram Protocol (UDP) destination ports, use the **ip rtp priority** command in interface configuration mode. To disable the strict priority queue, use the **no** form of this command.

ip rtp priority *starting-rtp-port-number* *port-number-range* *bandwidth*

no ip rtp priority

Syntax Description

<i>starting-rtp-port-number</i>	The starting RTP port number. The lowest port number to which the packets are sent. The port number can be a number from 2000 to 65,535.
<i>port-number-range</i>	The range of UDP destination ports. Number, when added to the <i>starting-rtp-port-number</i> argument, that yields the highest UDP port number. The range of UDP destination ports is from 0 to 16,383.
<i>bandwidth</i>	Maximum allowed bandwidth, in kbps. The maximum allowed bandwidth is from 0 to 2000.

Defaults

Disabled

Command Modes

Interface configuration

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

This command is most useful for voice applications, or other applications that are delay-sensitive.

This command extends and improves on the functionality offered by the **ip rtp reserve** command by allowing you to specify a range of UDP/RTP ports whose voice traffic is guaranteed strict priority service over any other queues or classes using the same output interface. Strict priority means that if packets exist in the priority queue, they are dequeued and sent first—that is, before packets in other queues are dequeued. We recommend that you use the **ip rtp priority** command instead of the **ip rtp reserve** command for voice configurations.

This command can be used in conjunction with either weighted fair queuing (WFQ) or class-based WFQ (CBWFQ) on the same outgoing interface. In either case, traffic matching the range of ports specified for the priority queue is guaranteed strict priority over other CBWFQ classes or WFQ flows; voice packets in the priority queue are always serviced first.

Remember the following guidelines when using the **ip rtp priority** command:

- When used in conjunction with WFQ, the **ip rtp priority** command provides strict priority to voice, and WFQ scheduling is applied to the remaining queues.
- When used in conjunction with CBWFQ, the **ip rtp priority** command provides strict priority to voice. CBWFQ can be used to set up classes for other types of traffic (such as Systems Network Architecture [SNA]) that need dedicated bandwidth and need to be treated better than best effort and not as strict priority; the nonvoice traffic is serviced fairly based on the weights assigned to the enqueued packets. CBWFQ can also support flow-based WFQ within the default CBWFQ class if so configured.

Remember the following guidelines when configuring the *bandwidth* argument:

- It is always safest to allocate to the priority queue slightly more than the known required amount of bandwidth, to allow room for network bursts.
- The IP RTP Priority admission control policy takes RTP header compression into account. Therefore, while configuring the *bandwidth* argument of the **ip rtp priority** command you need to configure only for the bandwidth of the compressed call. Because the *bandwidth* argument is the maximum total bandwidth, you need to allocate enough bandwidth for all calls if there will be more than one call.
- Configure a bandwidth that allows room for Layer 2 headers. The bandwidth allocation takes into account the payload plus the IP, UDP, and RTP headers but does not account for Layer 2 headers. Allowing 25 percent bandwidth for other overhead is conservative and safe.
- The sum of all bandwidth allocation for voice and data flows on an interface cannot exceed 75 percent of the total available bandwidth, unless you change the default maximum reservable bandwidth. To change the maximum reservable bandwidth, use the **max-reserved-bandwidth** command on the interface.

For more information on IP RTP Priority bandwidth allocation, refer to the section “IP RTP Priority” in the chapter “Congestion Management Overview” in the *Cisco IOS Quality of Service Solutions Configuration Guide*.

Examples

The following example first defines a CBWFQ configuration and then reserves a strict priority queue with the following values: a starting RTP port number of 16384, a range of 16383 UDP ports, and a maximum bandwidth of 40 kbps:

```
! The following commands define a class map:
class-map class1
match access-group 101
exit

! The following commands create and attach a policy map:
policy-map policy1
class class1
bandwidth 3000
queue-limit 30
random-detect
random-detect precedence 0 32 256 100
exit
interface Serial1
service-policy output policy1

! The following command reserves a strict priority queue:
ip rtp priority 16384 16383 40
```

Related Commands	Command	Description
	bandwidth (policy-map class)	Specifies or modifies the bandwidth allocated for a class belonging to a policy map.
	fair queue (WFQ)	Enables WFQ for an interface.
	frame-relay ip rtp priority	Reserves a strict priority queue on a Frame Relay PVC for a set of RTP packet flows belonging to a range of UDP destination ports.
	ip rtp reserve	Reserves a special queue for a set of RTP packet flows belonging to a range of UDP destination ports.
	max-reserved-bandwidth	Changes the percent of interface bandwidth allocated for CBWFQ, LLQ, and IP RTP Priority.
	policy-map	Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy.
	ppp multilink	Enables MLP on an interface and, optionally, enables dynamic bandwidth allocation.
	ppp multilink fragment-delay	Configures a maximum delay allowed for transmission of a packet fragment on an MLP bundle.
	ppp multilink interleave	Enables interleaving of RTP packets among the fragments of larger packets on an MLP bundle.
	priority	Gives priority to a class of traffic belonging to a policy map.
	service-policy	Attaches a policy map to an input interface or VC, or an output interface or VC, to be used as the service policy for that interface or VC.
	show policy-map	Displays the configuration of all classes for a specified service policy map or all classes for all existing policy maps.
	show queue	Displays the contents of packets inside a queue for a particular interface or VC.