# RSVP Infrastructure Commands

# authentication (RSVP)

To enter RSVP authentication mode, use the **authentication** command in global configuration mode, RSVP interface configuration mode, or RSVP neighbor XR Config mode. To remove authentication parameters in the applicable mode, use the **no** form of this command.

**authentication disable**
**no authentication**

## Syntax Description

| | |
|---|---|
| **disable** | Disable RSVP authentication. |

## Command Default

The default value is no authentication, which means that the feature is disabled.

## Command Modes

XR Config mode

RSVP interface configuration

RSVP neighbor configuration

## Command History

| Release | Modification |
|---|---|
| Release 7.11.1 | The **disable** keyword was introduced. |
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following example shows how to enter RSVP authentication configuration mode from global configuration mode:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp authentication
RP/0/RP0/CPU0:router(config-rsvp-auth)#
```

The following example shows how to activate the RSVP on an interface and enter RSVP authentication configuration mode:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface HundredGigE 0/0/0/3
RP/0/RP0/CPU0:router(config-rsvp-if)# authentication
```

```
RP/0/RP0/CPU0:router(config-rsvp-if-auth)#
```

The following example shows how to configure the RSVP neighbor with IP address 10.0.0.1 and enter neighbor authentication configuration mode:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp neighbor 10.0.0.1 authentication
RP/0/RP0/CPU0:router(config-rsvp-nbor-auth)#
```

The following example shows how to disable RSVP authentication:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface HundredGigE 0/0/0/3
RP/0/RP0/CPU0:router(config-rsvp-nbor-auth)#disable
```

# bandwidth mam (RSVP)

To configure RSVP bandwidth on an interface using the Maximum Allocation Model (MAM) bandwidth constraints model, use the **bandwidth mam** command in RSVP interface configuration mode. To return to the default behavior, use the **no** form of this command.

**bandwidth mam** [**percentage**] {*total-reservable-bandwidth* | **max-reservable-bw** *maximum-reservable-bw*} [*largest-reservable-flow* [**bc0** *reservable-bandwidth* ] [**bc1** *reservable-bw*]]
**no bandwidth mam**

## Syntax Description

| | |
|---|---|
| *total-reservable- bandwidth* | Total reservable bandwidth (in Kbps, Mbps or Gbps) that RSVP accepts for reservations on this interface. Range is 0 to 4294967295. |
| **max-reservable-bw** *maximum-reservable-bw* | Configures the maximum reservable bandwidth (in Kbps, Mbps or Gbps) that RSVP accepts for reservations on this interface. Range is 0 to 4294967295. |
| *largest-reservable-flow* | (Optional) Largest reservable flow (in Kbps, Mbps or Gbps) that RSVP accepts for reservations on this interface. Range is 0 to 4294967295. |
| **bc0** *reservable-bandwidth* | (Optional) Configures the total reservable bandwidth in the bc0 pool (in Kbps, Mbps or Gbps). |
| **bc1** *reservable-bw* | (Optional) Configures the total reservable bandwidth in the bc1 pool (in Kbps, Mbps or Gbps). |
| **percentage** | (Optional) Bandwidth as a percentage of physical link bandwidth. |

## Command Default

No default behavior or values.

## Command Modes

RSVP interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

Both the MAM and RDM models can be configured on a single interface to allow switching between each model.

**Note** Non-stop forwarding (NSF) is not guaranteed when the bandwidth constraint model is changed.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

| Task ID | Operations |
|---|---|
| ouni | read, write |

## Examples

The following example shows how to limit the total of all RSVP reservations on an interface:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface HundredGigE 0/0/0/3
RP/0/RP0/CPU0:router(config-rsvp-if)# bandwidth mam 7500
```

## Examples

The following example shows how to allocate a percentage of total bandwidth to bc0 and bc1 pools:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface HundredGigE 0/0/0/3
RP/0/RP0/CPU0:router(config-rsvp-if)# bandwidth mam percentage bc0 100 bc1 50
```

# bandwidth rdm (RSVP)

To configure RSVP bandwidth on an interface using the Russian Doll Model (RDM) bandwidth constraints model, use the **bandwidth rdm** command in RSVP interface configuration mode. To return to the default behavior, use the **no** form of this command.

**bandwidth rdm** [**percentage**] {*total-reservable-bw* | **bc0** *total-reservable-bw* | **global-pool** *total-reservable-bw*} [*largest-reservable-flow*] [**bc1** *reservable-bw*] [**sub-pool** *reservable-bw*]
**no bandwidth rdm**

## Syntax Description

| | |
|---|---|
| *total-reservable-bw* | Total reservable bandwidth (in Kbps, Mbps or Gbps). The default value is expressed in Kbps. |
| **bc0** *total-reservable-bw* | Reserves bandwidth in the bc0 pool (in Kbps, Mbps or Gbps). |
| **global-pool** | Reserves bandwidth in the global pool. |
| *largest-reservable-flow* | (Optional) Largest reservable flow (in Kbps, Mbps or Gbps). The default value is expressed in Kbps. |
| **bc1** | (Optional) Reserves bandwidth in the bc1 pool (in Kbps, Mbps or Gbps). |
| **sub-pool** | (Optional) Reserves bandwidth in the sub-pool. |
| *reservable-bandwidth* | Reservable bandwidth in the sub- and bc1 pools (in Kbps, Mbps or Gbps). The default value is expressed in Kbps. |
| **percentage** | (Optional) Bandwidth as a percentage of physical link bandwidth. |

## Command Default

No default behavior or values.

## Command Modes

RSVP interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

Both the MAM and RDM bandwidth constraint models support up two bandwidth pools.

Cisco IOS XR software provides global configuration when switching between bandwidth constraint models. Both models are configured on a single interface to allow switching between models.

**Note** Non-stop forwarding (NSF) is not guaranteed when the bandwidth constraint model is changed.

The **global pool** and **sub-pool** keywords are included in this command for backward compatibility with prestandard DS-TE. The **global pool** keyword is equivalent to the **bc0** keyword. The **sub-pool** keyword is equivalent to the **bc1** keyword.

RDM is the default bandwidth constraint model used in both pre-standard and IETF mode.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |
| ouni | read, write |

## Examples

The following example shows how to limit the total of all RSVP reservations on an interface to 7500 kbps, and allows each single flow to reserve no more than 1000 kbps:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface HundredGigE 0/0/0/3
RP/0/RP0/CPU0:router(config-rsvp-if)# bandwidth rdm 7500 1000
```

## Examples

The following example shows how to allocate a percentage of total bandwidth to bc0 and bc1 pools:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface HundredGigE 0/0/0/3
RP/0/RP0/CPU0:router(config-rsvp-if)# bandwidth rdm percentage 100 bc0 100 bc1 50
```

# bandwidth (RSVP)

To configure RSVP bandwidth on an interface using prestandard DS-TE mode, use the **bandwidth** command in RSVP interface XR Config mode. To reset the RSVP bandwidth on that interface to its default value, use the **no** form of this command.

**bandwidth** [**percentage**] [*total-reservable-bandwidth* [*largest-reservable-flow*] [**sub-pool** *reservable-bw*]] [**global-pool** *bandwidth* [**sub-pool** *reservable-bw*]] [**bc0** *bandwidth* [**bc1** *reservable-bw*]]
**no bandwidth**

## Syntax Description

| | |
|---|---|
| *total-reservable-bandwidth* | (Optional) Total reservable bandwidth (in Kbps, Mbps or Gbps) that RSVP accepts for reservations on this interface. Range is 0 to 4294967295. |
| *largest-reservable-flow* | (Optional) Largest reservable flow (in Kbps, Mbps or Gbps) that RSVP accepts for reservations on this interface. Range is 0 to 4294967295. |
| **sub-pool** *reservable-bw* | (Optional) Configures the total reservable bandwidth in the sub-pool (in Kbps, Mbps, or Gbps). Range is 0 to 4294967295. |
| **bc0** *bandwidth* | (Optional) Configures the total reservable bandwidth in the bc0 pool (in Kbps, Mbps or Gbps). The default is Kbps. Range is 0 to 4294967295. |
| **bc1** *reservable-bw* | (Optional) Configures the total reservable bandwidth in the bc1 pool (in Kbps, Mbps or Gbps). |
| **global-pool** *bandwidth* | (Optional) Configures the total reservable bandwidth in the global-pool. Range is 0 to 4294967295 Kbps. |
| **percentage** | (Optional) Bandwidth as a percentage of physical link bandwidth. |

## Command Default

*sub-pool-bw*: 0

**Note** If the command is entered without the optional arguments, the total bandwidth is set to 75 percent of the intrinsic bandwidth of the interface. (If the interface has zero intrinsic bandwidth, none are reserved.)

## Command Modes

RSVP interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

RSVP is enabled either using the **rsvp interface** command or when MPLS is configured on the interface. In addition, there are other instances in which RSVP is enabled automatically; for example, when an RSVP message is received on an interface that is not configured under RSVP or MPLS (such as out-of-band signaling for an Optical User Network Interface application).

If RSVP reservation messages are received on an interface different from the one through which the corresponding Path message was sent out, the interfaces are adjusted such that all resource reservations, such as bandwidth, are done on the outgoing interface of the Path message.

Prestandard DS-TE uses the Cisco proprietary mechanisms for RSVP signaling and IGP advertisements. This DS-TE mode does not interoperate with third-party vendor equipment. Note that prestandard DS-TE is enabled only after configuring the sub-pool bandwidth values on MPLS-enabled interfaces.

**Note** You can also configure RSVP bandwidth on an interface using IETF DS-TE mode. This mode supports multiple bandwidth constraint models, including the Russian Doll Model (RDM) and the Maximum Allocation Model (MAM) both with two bandwidth pools.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |
| ouni | read, write |

## Examples

The following example shows how to limit the total of all RSVP reservations on HundredGigE interface 0/0/0/3 to 5000 Kbps:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface HundredGigE 0/0/0/3
RP/0/RP0/CPU0:router(config-rsvp-if)# bandwidth 5000
```

## Examples

The following example shows how to allocate a percentage of total bandwidth to bc0 and bc1 pools:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface HundredGigE 0/0/0/3
RP/0/RP0/CPU0:router(config-rsvp-if)# bandwidth percentage bc0 100 bc1 50
```

# clear rsvp authentication

To eliminate RSVP security association (SA) before the lifetime expires, use the **clear rsvp authentication** command in XR EXEC mode.

**clear rsvp authentication** [*type interface-path-id*] [**destination** *IP address* ] [**source** *IP address*]

## Syntax Description

| | |
|---|---|
| *type* | (Optional) Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Physical interface or a virtual interface.<br>**Note** Use the **show interfaces** command to see a list of all possible interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **destination** *IP address* | (Optional) Eliminates the RSVP security associations (SA) before their lifetimes expire. All SAs with this destination IP address are cleared. |
| **source** *IP address* | (Optional) Eliminates the RSVP security associations (SA) before their lifetimes expire. All SAs with this source IP address are cleared. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

Use the **clear rsvp authentication** command for the following reasons:

- To eliminate security associations before their lifetimes expire
- To free up memory
- To resolve a problem with a security association being in an indeterminate state

You can delete all RSVP security associations if you do not enter an optional filter (interface, source, or destination IP address).

If you delete a security association, it is recreated as needed.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | execute |

## Examples

The following example shows how to clear each SA:

```
RP/0/RP0/CPU0:router# clear rsvp authentication
```

The following example shows how to clear each SA with the destination address 10.0.0.1:

```
RP/0/RP0/CPU0:router# clear rsvp authentication destination 10.0.0.1
```

The following example shows how to clear each SA with the source address 172.16.0.1:

```
RP/0/RP0/CPU0:router# clear rsvp authentication source 172.16.0.1
```

The following example shows how to clear each SA with the interface:

```
RP/0/RP0/CPU0:router# clear rsvp authentication HundredGigE 0/0/0/3
```

The following example shows how to clear each SA on the interface, destination address 10.0.0.1, and source address 172.16.0.1:

```
RP/0/RP0/CPU0:router# clear rsvp authentication HundredGigE 0/0/0/3 destination 10.0.0.1
source 172.16.0.1
```

# clear rsvp counters all

To clear (set to zero) all RSVP message and event counters that are being maintained by the router, use the **clear rsvp counters all** command in XR EXEC mode.

**clear rsvp counters all** [*type interface-path-id*]

## Syntax Description

| | |
|---|---|
| *type* | (Optional) Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Physical interface or a virtual interface.<br>**Note** Use the **show interfaces** command to see a list of all possible interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following example shows how to clear all message and event counters:

```
RP/0/RP0/CPU0:router# clear rsvp counters all
```

# clear rsvp counters authentication

To eliminate RSVP counters for each security association (SA), use the **clear rsvp counters authentication** command in XR EXEC mode.

**clear rsvp counters authentication** [*type interface-path-id*] [**destination** *IP address* ][**source** *IP address* ]

## Syntax Description

| | |
|---|---|
| *type* | (Optional) Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Physical interface or a virtual interface.<br>**Note** Use the **show interfaces** command to see a list of all possible interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **destination** *IP address* | (Optional) Eliminates authentication-related statistics for each security association (SA) with this destination IP address. |
| **source** *IP address* | (Optional) Eliminates authentication-related statistics for each security association (SA) with this source IP address. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | execute |

## Examples

The following example shows how to clear authentication counters for each SA:

```
RP/0/RP0/CPU0:router# clear rsvp counters authentication
```

The following example shows how to clear authentication counters for each SA with the destination address 10.0.0.1:

```
RP/0/RP0/CPU0:router# clear rsvp counters authentication destination 10.0.0.1
```

The following example shows how to clear authentication counters for each SA with the source address 172.16.0.1:

```
RP/0/RP0/CPU0:router# clear rsvp counters authentication source 172.16.0.1
```

The following example shows how to clear authentication counters for each SA with an interface.

```
RP/0/RP0/CPU0:router# clear rsvp counters authentication HundredGigE 0/0/0/3
```

The following example shows how to clear authentication counters for each SA on an interface, destination address 10.0.0.1, and source address 172.16.0.1:

```
RP/0/RP0/CPU0:router# clear rsvp counters authentication HundredGigE 0/0/0/3 destination
10.0.0.1 source 172.16.0.1
```

# clear rsvp counters chkpt

To clear RSVP checkpoint counters, use the **clear rsvp counters chkpt** command in XR EXEC mode.

**clear rsvp counters chkpt**

## Syntax Description

This command has no arguments or keywords.

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following example shows how to clear all message and event counters:

```
RP/0/RP0/CPU0:router# clear rsvp counters chkpt
```

# clear rsvp counters events

To clear (set to zero) all RSVP event counters that are being maintained by the router, use the **clear rsvp counters events** command in XR EXEC mode.

**clear rsvp counters events** [*type interface-path-id*]

## Syntax Description

| | |
|---|---|
| *type* | (Optional) Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Physical interface or a virtual interface.<br>**Note** Use the **show interfaces** command to see a list of all possible interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

Use the **clear rsvp counters events** command to set all RSVP event counters to zero.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following example shows how to clear all event counters:

```
RP/0/RP0/CPU0:router# clear rsvp counters events
```

# clear rsvp counters messages

To clear (set to zero) all RSVP message counters that are being maintained by the router, use the **clear rsvp counters messages** command in XR EXEC mode.

**clear rsvp counters messages** [*type interface-path-id*]

## Syntax Description

| | |
|---|---|
| *type* | (Optional) Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Physical interface or a virtual interface.<br>**Note** Use the **show interfaces** command to see a list of all possible interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

Use the **clear rsvp counters messages** command to set all RSVP message counters to zero.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following example shows how to set all RSVP message counters for an interface to zero:

```
RP/0/RP0/CPU0:router# clear rsvp counters messages HundredGigE0/0/0/3
```

# clear rsvp counters oor

To clear internal RSVP counters on out of resources (OOR) events, use the **clear rsvp counters oor** command in XR EXEC mode.

**clear rsvp counters oor** [*type interface-path-id*]

## Syntax Description

| | |
|---|---|
| *type* | (Optional) Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Physical interface or a virtual interface.<br>**Note** Use the **show interfaces** command to see a list of all possible interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

Use the **clear rsvp counters oor** command to set RSVP OOR counters to zero.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following example show how to clear all RSVP message counters for HundredGigE interface 0/0/0/3 to zero:

```
RP/0/RP0/CPU0:router# clear rsvp counters oor HundredGigE0/0/0/3
```

# clear rsvp counters prefix-filtering

To clear internal prefix-filtering related RSVP counters, use the **clear rsvp counters prefix-filtering** command in XR EXEC mode.

**clear rsvp counters prefix-filtering** {**interface** [*type* *interface-path-id*] | **access-list** [*aclname*]}

## Syntax Description

| | |
|---|---|
| **interface** | Clears RSVP prefix-filtering counters for all interfaces. |
| *type* | (Optional) Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Physical interface or a virtual interface.<br>**Note** Use the **show interfaces** command to see a list of all possible interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **access-list** | Clears RSVP prefix-filtering counters for access control list. |
| *aclname* | (Optional) Name of the access list. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

Use the **clear rsvp counters prefix-filtering** command to set RSVP prefix-filtering related RSVP counters to zero.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following example shows how to set all RSVP message counters for HundredGigE interface 0/0/0/3 to zero:

```
RP/0/RP0/CPU0:router# clear rsvp counters prefix-filtering interface HundredGigE0/0/0/3
```

The following example shows how to set all RSVP prefix-filtering counters for access-list banks to zero:

```
RP/0/RP0/CPU0:router# clear rsvp counters prefix-filtering access-list banks
```

# key-source key-chain (RSVP)

To specify the source of the key information to authenticate RSVP messages, use the **key-source key-chain** command in the appropriate RSVP authentication configuration mode. To remove the key source from the appropriate RSVP authentication configuration mode, use the **no** form of this command.

**key-source key-chain** *key-chain-name*
**no key-source key-chain** *key-chain-name*

## Syntax Description

| | |
|---|---|
| *key-chain-name* | Name of the keychain. The maximum number of characters is 32. |

## Command Default

The default value is none, which means that the key source is not specified.

## Command Modes

RSVP authentication configuration

RSVP interface authentication configuration

RSVP neighbor authentication configuration

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

- RSVP authentication is enabled regardless of whether or not the specified keychain exists or has no available keys to use. If the specified keychain does not exist or there are no available keys in the keychain, RSVP authentication processing fails.
- The **key-source key-chain** command does not create a keychain but just specifies which keychain to use. You must configure a keychain first.
- The **no key-source key-chain** command does not necessarily disable the authentication.
- RSVP authentication supports only keyed-hash message authentication code (HMAC)-type algorithms.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following example shows that the source of the key information is specified for the keychain mpls-keys in RSVP authentication configuration mode:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp authentication
RP/0/RP0/CPU0:router(config-rsvp-auth)# key-source key-chain mpls-keys
```

The following example shows that the source of the key information is specified for the keychain mpls-keys for a HundredGigE interface in RSVP authentication configuration mode:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface HundredGigE 0/0/0/3
RP/0/RP0/CPU0:router(config-rsvp-if)# authentication
RP/0/RP0/CPU0:router(config-rsvp-if-auth)# key-source key-chain mpls-keys
```

The following example shows that the source of the key information is specified for the keychain mpls-keys in RSVP neighbor authentication configuration mode:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp neighbor 10.0.0.1 authentication
RP/0/RP0/CPU0:router(config-rsvp-nbor-auth)# key-source key-chain mpls-keys
```

# life-time (RSVP)

To control how long RSVP maintains idle security associations with other trusted RSVP neighbors, use the **life-time** command in the appropriate RSVP authentication configuration mode. To disable the lifetime setting, use the **no** form of this command.

**life-time** *seconds*
**no life-time** *seconds*

## Syntax Description

| | |
|---|---|
| *seconds* | Length of time, in seconds, that RSVP maintains security associations with other trusted RSVP neighbors. Range is 30 to 86400. |

## Command Default

*seconds*: 1800 (30 minutes)

## Command Modes

RSVP authentication configuration

RSVP interface authentication configuration

RSVP neighbor authentication configuration

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

Use the **life-time (RSVP)** command to indicate when to end idle security associations with RSVP trusted neighbors.

By setting a larger lifetime, the router remembers the state for a long period time which provides better protection against a replay attack.

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

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following example shows how to configure a lifetime of 2000 seconds for each SA in RSVP authentication configuration mode:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp authentication
RP/0/RP0/CPU0:router(config-rsvp-auth)# life-time 2000
```

The following example shows how to configure a lifetime of 2000 seconds for each SA in RSVP neighbor authentication configuration mode:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp neighbor 10.0.0.1 authentication
RP/0/RP0/CPU0:router(config-rsvp-nbor-auth)# life-time 2000
```

The following example shows how to configure a lifetime of 2000 seconds for each SA in RSVP interface authentication configuration mode:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface HundredGigE 0/0/0/3
RP/0/RP0/CPU0:router(config-rsvp-if)# authentication
RP/0/RP0/CPU0:router(config-rsvp-if-auth)# life-time 2000
```

# mpls traffic-eng lsp-oor

To set LSP out-of-resource (OOR) parameters, use the **mpls traffic-eng lsp-oor** command in XR Config mode. To remove LSP OOR parameter settings, use the **no** form of this command.

**mpls traffic-eng lsp-oor** [ { **green** | **red** | **yellow** } **action** { **accept reopt-lsp** | **admit lsp-min-bw** *value* | **flood** { **available-bw** *value* | **te-metric penalty** *value* } } | { **yellow** | **red** } { **transit-all** | **transit-unprotected** } **threshold** *value* | **green recovery-duration** *minutes* ]

**no mpls traffic-eng lsp-oor** [ { **green** | **red** | **yellow** } **action** { **accept reopt-lsp** | **admit lsp-min-bw** *value* | **flood** { **available-bw** *value* | **te-metric penalty** } } | { **yellow** | **red** } { **transit-all** | **transit-unprotected** } **threshold** | **green recovery-duration** ]

## Syntax Description

| | |
|---|---|
| {**green**\|**red**\|**yellow**} | (Optional) Specifies a color option for identifying specific actions noted with the **action** keyword.<br><br>Here, *green* signifies *normal* state, *red* signifies *major* state, and *yellow* signifies *minor* state. |
| **action** {**accept reopt-lsp**\|**admit lsp-min-bw** *value*\|**flood** {**available-bw** *value*\|**te-metric penalty** *value*}} | (Optional) Specifies one of the three actions for the selected state:<br><br>• **accept reopt-lsp** – Accepts a reoptimized LSP sharing the same link in the selected state as the current LSP. If not enabled, reoptimized LSPs are rejected.<br><br>• **admit lsp-min-bw** *value* – Accept LSPs with a bandwidth that is at least equal to the specified bandwidth. The default value is 0.<br><br>• **flood te-metric penalty** *value* – Adds a penalty value to the TE metric of the links in the specified state. This metric is flooded for all links on the router. The default value is 0.<br><br>• **flood available-bw** *value* – Specifies the percentage of available bandwidth for all links. The default value is 100%. |

| | |
|---|---|
| {**yellow**\|**red**}{**transit-all**\|**transit-protected**} **threshold** *value* | (Optional) Specifies a threshold value for mid-point (or transit) LSRs, for the yellow and red color options.<br>• **transit-all** – Specifies that the threshold value be applied for all mid-point routers.<br>• **transit-unprotected** – Specifies that the threshold value be applied for unprotected mid-point routers.<br>• **threshold** *value* – Specifies the threshold value. |
| **green recovery-duration** *minutes* | (Optional) Specifies the time duration for an LSP action in the *green* state, after recovery. The default value is 0 minutes. |

**Command Default**

LSP OOR parameters are disabled.

**Command Modes**

Global Configuration

**Command History**

| Release | Modification |
|---|---|
| Release 7.3.2 | This command was introduced. |

**Usage Guidelines**

Use the **mpls traffic-eng lsp-oor .. action flood available-bw** *value* command form to lower the available bandwidth on the link, potentially reducing the number of states that would be possible to set up over the link.

Use the **mpls traffic-eng lsp-oor .. action flood te-metric penalty** *value* command form to add to the flooded TE metric (in the MPLS-TE topology). This serves as a deterrent for LERs to set up LSPs over this link.

Use the **mpls traffic-eng lsp-oor** .. **action admit lsp-min-bw** *value* command form to admit only new LSPs with signaled bandwidth that exceeds the bandwidth value. This restricts the number of new transit LSPs to only a few high bandwidth LSPs.

Use the **mpls traffic-eng lsp-oor** .. **action accept reopt-lsp** command form to recover the condition when LSPs run into *Yellow* or *Red* states, by allowing existing LSPs to re-optimize.

Use the **mpls traffic-eng lsp-oor** .. **green recovery-duration** *minutes* command form to determine how long the actions are taken in the LSP OOR *Green* state after recovery. In other words, moving from yellow state to green state or red state to green state.

The following example shows how to configure the time duration for an LSP action in the *green* state, after recovery

```
Router# configure
Router(config)# mpls traffic-eng lsp-oor green recovery-duration 10
Router(config)# commit
Router(config)# end
```

The following example shows the output for the **show mpls traffic-eng lsp-oor summary** command. The main counters track the current OOR state, OOR thresholds, transitions, and the number of LSPs rejected due to OOR.

```
Router# show mpls traffic-eng lsp-oor summary

Total Transit LSPs: 5001
Total Transit Unprotected LSPs: 0
LSP OOR Status: Yellow; Changed last at: Wed May 15 17:05:48 2019
LSP OOR Green State Parameters:
  Available Bandwidth percentage: 100%
  TE Metric Penalty: 0
  Minimum LSP Size: 0 kbps
  Accept Reopt: FALSE
  Transition duration: 0 minutes
  Statistics:
    Transitions 0; LSPs accepted 5001, rejected 0
    Reopt accepted 0, rejected 0
LSP OOR Yellow State Parameters:
  Available Bandwidth percentage: 0%
  TE Metric Penalty: 0
  Minimum LSP Size: 10000 kbps
  Accept Reopt: TRUE
  Transit LSP Threshold: 5000
  Transit Unprotected LSP Threshold: No limit
  Statistics:
    Transitions 1; LSPs accepted 0, rejected 999
    Reopt accepted 0, rejected 0
LSP OOR Red State Parameters:
  Available Bandwidth percentage: 0%
  TE Metric Penalty: 0
  Minimum LSP Size: 10000 kbps
  Accept Reopt: FALSE
  Transit LSP Threshold: 10000
  Transit Unprotected LSP Threshold: No limit
  Statistics:
    Transitions 0; LSPs accepted 0, rejected 0
    Reopt accepted 0, rejected 0
```

# rsvp

To enable functionality for Resource Reservation Protocol (RSVP) and enter RSVP configuration commands, use the **rsvp** command in XR Config mode. To return to the default behavior, use the **no** form of this command.

**rsvp**
**no rsvp**

## Syntax Description

This command has no keywords or arguments.

## Command Default

No default behavior or values

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following example shows how to enable RSVP functionality and enter the sub-mode for RSVP configuration commands:

```
RP/0/RP0/CPU0:router(config)# rsvp
RP/0/RP0/CPU0:router(config-rsvp)#
```

# rsvp interface

To configure RSVP on an interface, use the **rsvp interface** command in XR Config mode. To disable RSVP on that interface, use the **no** form of this command.

**rsvp interface** *type interface-path-id*
**no rsvp interface** *type interface-path-id*

## Syntax Description

| | |
|---|---|
| *type* | Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Physical interface or a virtual interface.<br>**Note** Use the **show interfaces** command to see a list of all possible interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |

## Command Default

RSVP is enabled by default on an interface under the following conditions. (Enabling RSVP on an interface means that interface can be used by RSVP to send and receive RSVP messages).

- RSVP is configured on that interface using the **rsvp interface** command.
- MPLS is configured on that interface.

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

When RSVP is enabled on an interface by any of the three methods mentioned in the above section, the default bandwidth is 0. Use the bandwidth command in RSVP interface configuration mode to configure the bandwidth on an interface.

If the interface bandwidth is 0, RSVP can be used only to signal flows that do not require bandwidth on this interface.

The **rsvp interface** command enables the RSVP interface configuration mode.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following example shows how to enable the RSVP interface configuration mode and to enable RSVP on this interface with 0 bandwidth:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface HundredGigE 0/0/0/3
```

# rsvp neighbor

To specify an RSVP neighbor, use the **rsvp neighbor** command in XR Config mode. To deactivate authentication for a neighbor, use the **no** form of this command.

**rsvp neighbor** *IP-address* **authentication**
**no rsvp neighbor** *IP-address* **authentication**

## Syntax Description

| | |
|---|---|
| *IP-address* | IP address of the neighbor. A single IP address of a specific neighbor; usually one of the neighbor's physical or logical (loopback) interfaces. |
| **authentication** | Configures RSVP authentication parameters. |

## Command Default

No default values or behaviors

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

RSVP neighbor configuration mode can be used only if you want to configure authentication for a particular neighbor.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following example shows how to enter RSVP neighbor authentication configuration mode for IP address 10.0.0.1:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp neighbor 10.0.0.1 authentication
RP/0/RP0/CPU0:router(config-rsvp-nbor-auth)#
```

# show rsvp request

To list all the requests that RSVP knows about on a router, use the **show rsvp request** command in XR EXEC mode.

**show rsvp request** [**destination** *IP-address*] [**detail**] [**dst-port** *port-num*] [**session-type** { **lsp-p2p** }] [**source** *IP-address*] [**src-port** *port-num*]

## Syntax Description

| | |
|---|---|
| **detail** | (Optional) Displays multiline status for each path. If this keyword is not specified, only a single-line table entry is displayed. |
| **destination** *IP-address* | (Optional) Displays the entries that match the specified address. |
| **dst-port** *port-num* | (Optional) Displays destination port and tunnel information. |
| **session-type** | (Optional) Displays the entries that match the specified session type. |
| **lsp-p2p** | Displays the entries that are used for P2P sessions. |
| **source** *IP-address* | (Optional) Displays source address information. |
| **src-port** *port-num* | (Optional) Displays port and LSP ID information. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

This command displays information about upstream reservations only; that is, reservations being sent to upstream hops. Information about downstream reservations (that is, incoming or locally created reservations) is available using the **show rsvp reservation** command.

Reservations are displayed in ascending order of destination IP address, destination port, source IP address, and source port.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following is sample output from the **show rsvp request** command:

```
RP/0/RP0/CPU0:router# show rsvp request

        Dest Addr DPort     Source Addr SPort Pro   OutputIF          Sty Serv Rate Burst
```

```
  ---------------- ----- ---------------- ----- --- ----------          --- ---- ---- -----

     192.168.40.40 2001     192.168.67.68     2   0  HundredGigE 0/0/0/3  SE LOAD    0
1K
```

The following is sample output from the **show rsvp request detail** command, which displays detailed information about all requests in the router. Requests are reservation states for the reservation messages sent upstream:

```
RP/0/RP0/CPU0:router# show rsvp request detail

  REQ:  IPv4-LSP Session addr: 192.168.40.40. TunID: 2001. LSPId: 2.
   Source addr: 192.168.67.68. ExtID: 192.168.67.68.
   Output interface: HundredGigE 0/0/0/3. Next hop: 192.168.67.68 (lih: 0x19700001).
   Flags: Local Receiver.
   Style: Shared-Explicit. Service: Controlled-Load.
   Rate: 0 bits/sec. Burst: 1K bytes. Peak: 0 bits/sec.
   MTU min: 0, max: 500 bytes.
   Policy:  Forwarding. Policy source(s): MPLS/TE.
   Number of supporting PSBs: 1
   Destination Add DPort       Source Add SPort Pro     Input IF            Rate Burst Prot

    192.168.40.40  2001        192.168.67.68 2   0    HundredGigE 0/0/0/3    0   1K    Off
   Number of supporting RSBs: 1
   Destination Add DPort       Source Add SPort Pro   Input IF Sty Serv Rate Burst
     192.168.40.40  2001       10.66.67.68  2   0      None SE LOAD     0    1K
```

This table describes the significant fields shown in the display.

***Table 1: show rsvp request detail Command Field Descriptions***

| Field | Description |
|---|---|
| Number of supporting PSBs | Number of senders for this session (typically, 1). |
| Number of supporting RSBs | Number of reservations per session (typically, 1). |
| Policy | Admission control status. |
| Policy source | Entity performing the admission control. |

# show rsvp authentication

To display the database for the security association that RSVP has established with other RSVP neighbors, use the **show rsvp authentication** command in XR EXEC mode.

**show rsvp authentication** [*type interface-path-id*] [**destination** *IP-address*] [**detail**] [**mode** {**receive** | **send**}] [**neighbor** *IP-address*] [**source** *IP-address*]

## Syntax Description

| | |
|---|---|
| *type* | (Optional) Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Physical interface or a virtual interface.<br>**Note** Use the **show interfaces** command to see a list of all possible interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **destination** *IP-address* | (Optional) Displays the database for the security association (SA) for the destination IP address. The *IP address* argument is the IP address of the destination address. |
| **detail** | (Optional) Displays additional information about RSVP security SAs. |
| **mode** | (Optional) Specifies the SA type. An SA is used to authenticate either incoming (receive) or outgoing (send) messages. |
| **receive** | Displays SAs for incoming messages. |
| **send** | Displays SAs for outgoing messages. |
| **neighbor** *IP-address* | (Optional) Displays the RSVP authentication information for the neighbor IP address. The *IP-address* argument is the IP address of the neighbor. For the send SA, the neighbor address is the destination address. For receive, the neighbor address is the source address. |
| **source** *IP-address* | (Optional) Displays the database for the SA for the source IP address. The *IP-address* argument is the IP address of the source address. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read |

## Examples

The following sample output displays information for RSVP authentication:

```
RP/0/RP0/CPU0:router# show rsvp authentication

  Codes: S - static, G - global, N - neighbor, I -interface, C - chain

  Source Address  Dest Address    Interface             Mode Key-Source Key-ID Code
  10.0.0.1          10.0.0.2      HundredGigE 0/0/0/3   Send mpls-keys    1    SGC
  10.0.0.2          10.0.0.1      HundredGigE 0/0/0/3   Recv mpls-keys    1    SGC
```

This table describes the significant fields shown in the display.

***Table 2: show rsvp authentication Command Field Descriptions***

| Field | Description |
|---|---|
| Source Address | IP address of the sender. For Send mode, this is the local address (either the address of the Interface field or the local router ID). For Recv mode, this is the address of the RSVP neighbor. |
| Dest Address | IP address of the receiver. For Send mode, this is the address of the RSVP neighbor. For Recv mode, this is the local address (either the address of the Interface field or the local router ID). |
| Interface | Name of the interface over which the security association is being maintained. |
| Mode | Direction of the association for the following mode types:<br>**Send**<br>Authenticates messages that you forward.<br>**Recv**<br>Authenticates messages that you receive. |
| Key-Source | Key source identification string that is currently set to the configured keychain name. |
| Key-ID | The last successful key ID that is used for authentication and maps to the keychain ID configuration. If the value is too large to fit into the column, it is truncated and a (..) suffix is appended. Use the detail mode to see the non-truncated key ID. |

| Field | Description |
|---|---|
| Code | Code field has the following terms:<br>**Static**<br>Key is static and configured.<br>**Global**<br>Key is global-based.<br>**Neighbor**<br>Key is neighbor-based.<br>**Interface**<br>Key is interface-based.<br>**Chain**<br>Key is part of a keychain. |

The following sample output shows detailed information about a Send mode SA that is followed by a Receive mode SA:

```
RP/0/RP0/CPU0:router# show rsvp authentication detail

  RSVP Authentication Information:
    Source Address:         10.0.0.1
    Destination Address:    10.0.0.2
    Neighbour Address:      10.0.0.2
    Interface:              HundredGigE 0/0/0/3
    Direction:              Send
    LifeTime:               1800 (sec)
    LifeTime left:          1305 (sec)
    KeyType:                Static Global KeyChain
    Key Source:             name1
    Key Status:             No error
    KeyID:                  1
    Digest:                 HMAC MD5 (16)
    Challenge:              Not supported
    TX Sequence:            5023969459702858020 (0x45b8b99b00000124)
    Messages successfully authenticated:    245
    Messages failed authentication:         0

  Receive Errors:
       Incomplete security association:      0
       Missing INTEGRITY object:             0
       Incorrect digest:                     0
       Digest type mismatch:                 0
       Duplicate sequence number:            0
       Out-of-range sequence number:         0
       Invalid message format:               0
```

This table describes the significant fields shown in the display.

***Table 3: show rsvp authentication detail Command Field Descriptions***

| Field | Description |
|---|---|
| Source Address | IP address of the sender. For Send mode, this is the local address (either the address of the Interface field or the local router ID). For Recv mode, this is the address of the RSVP neighbor. |
| Destination Address | IP address of the receiver. For Send mode, this is the address of the RSVP neighbor. For Recv mode, this is the local address (either the address of the Interface field or the local router ID). |
| Neighbor Address | IP address of the RSVP neighbor with which the security association is being maintained. |
| Interface | Name of the interface over which the security association is being maintained. |
| Direction | Direction of the association for the following mode types:<br>**Send**<br>Authenticates messages that you forward.<br>**Recv**<br>Authenticates messages that you receive. |
| LifeTime | Configured expiration timer value. |
| LifeTime left | Number of seconds until the expiration timer expires. |
| KeyType | Keys that are used:<br>**Static**<br>Key is static and configured.<br>**Global**<br>Key is global-based.<br>**Neighbor**<br>Key is neighbor-based.<br>**Interface**<br>Key is interface-based.<br>**Chain**<br>Key is part of a keychain. |
| Key-Source | Key source identification string that is currently set to the configured keychain name. |
| Key Status | Last status reported from the key source. |

| Field | Description |
|---|---|
| Key-ID | Last successful key ID that is used for authentication and that maps to the keychain ID configuration. If the value is too large to fit into the column, it is truncated and a (..) suffix is appended. (Use the detail mode to see the non-truncated key ID.) |
| Digest | Digest algorithm that is used. The algorithms are either HMAC-MD5 or HMAC-SHA1. |
| Challenge | Current challenge status reported. |
| Tx Sequence | Last sequence number that was sent. |
| Messages successfully authenticated | Number of messages authenticated by using this SA. |
| Messages failed authentication | Number of messages that failed authentication using this SA. |
| Sequence Window Size | Maximum configured RX sequence number window. |
| Sequence Window Count | Currently used size of the RX sequence number window. |
| Incomplete security association | Number of messages that are dropped due to a key failure. |
| Incorrect digest | Number of messages that are dropped due to an incorrect digest. |
| Digest type mismatch | Number of messages that are dropped due to an incorrect digest length, which implies an algorithm mismatch. |
| Duplicate sequence number | Number of messages that are dropped due to a duplicate sequence number. |
| Out-of-range sequence number | Number of messages that are dropped due to a sequence number range (window-size) checking. |
| Invalid message format | Number of messages that are dropped due to formatting errors, such as incorrect objects. |

# show rsvp counters

To display internal RSVP counters, use the **show rsvp counters** command in XR EXEC mode mode.

**show rsvp counters** {**messages** [*type interface-path-id* | **summary** ] | **events** | **database**}

## Syntax Description

| | |
|---|---|
| **messages** | Displays a historical count of the number of messages RSVP has received and sent on each interface along with a summation. |
| *type* | (Optional) Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Physical interface or a virtual interface.<br>**Note** Use the **show interfaces** command to see a list of all possible interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **summary** | (Optional) Displays the aggregate counts of all interfaces. |
| **events** | Displays the number of states expired for lack of refresh and a count of received No Acknowledgements (NACKs). |
| **database** | Displays counters on RSVP database, including number of paths, session, and so on. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

In message counters, bundle messages are counted as single bundle messages. The component messages are not counted separately.

The **messages** keyword shows the counters for all the interfaces. In addition, the aggregate summary is shown by using both the **messages** and **summary** keywords.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following is sample output from the **show rsvp counters messages** command for HundredGigE 0/0/0/3:

```
RP/0/RP0/CPU0:router# show rsvp counters messages HundredGigE 0/0/0/3

    HundredGigE0/0/0/3      Recv     Xmit  Recv                Xmit
      Path                  24          1   Resv                   0          0
      PathError              0          0   ResvError              0          0
      PathTear               5          1   ResvTear               0          0
      ResvConfirm            0          0   Ack                   34          0
      Bundle                 0              Hello                  0          0
      SRefresh           10118          0   OutOfOrder             0
      Retransmit                        0   Rate Limited                      0
```

This table describes the significant fields shown in the display.

***Table 4: show rsvp counters messages Command Field Descriptions***

| Field | Description |
|---|---|
| Path | Number of Path messages sent downstream or received from an upstream node. |
| PathError | Number of PathError messages received from a downstream neighbor or sent to an upstream neighbor. |
| PathTear | Number of PathTear messages sent downstream, or messages received, from upstream neighbors. |
| ResvConfirm | Number of ResvConfirm messages received from an upstream neighbor or sent to a downstream neighbor. |
| Bundle | Number of Bundle messages containing RSVP messages sent and received by the neighbor. |
| SRefresh | Number of Summary Refresh messages sent to and received by a neighbor to refresh the path and reservation states. |
| Retransmit | Number of messages retransmitted to ensure reliable messaging (related to refresh reduction). |
| Resv | Number of Reservation messages received from a downstream neighbor or sent to an upstream neighbor to reserve resources. |
| ResvError | Number of Reservation Error messages received from a upstream neighbor or sent to a downstream neighbor. |
| ResvTear | Number of Reservation Tear messages received from a downstream neighbor or sent to an upstream neighbor to tear down RSVP flows. |
| Ack | Number of Acknowledgement messages sent and received by a neighbor acknowledging receipt of a message. |
| Hello | Number of Hello messages sent to and received by a neighbor. |
| OutOfOrder | Number of messages received that are out of order. |
| Rate Limited | Number of RSVP packets affected by rate limiting. |

The following is sample output from the **show rsvp counters database** command:

```
RP/0/RP0/CPU0:router# show rsvp counters database

  Sessions: 0
  Locally created and incoming paths: 0
  Outgoing paths: 0
  Locally created and incoming Reservations: 0
  Outgoing Reservations: 0
  Interfaces: 4
```

This table describes the significant fields shown in the display.

***Table 5: show rsvp counters database Command Field Descriptions***

| Field | Description |
|---|---|
| Sessions | RSVP sessions. |
| Locally created and incoming paths | Path states created by :<br>• A local application on the node.<br>• Path message received from the network. |
| Outgoing paths | Outgoing path states. |
| Locally created and incoming Reservations | Reservations created by :<br>• A local application on the node.<br>• Path message received from the network. |
| Outgoing Reservations | Outgoing reservation (request) states. |
| Interfaces | Known RSVP interfaces. |

# show rsvp counters oor

To display internal RSVP counters on out of resources (OOR) events, use the **show rsvp counters oor** command in XR EXEC mode.

**show rsvp counters oor** [*type interface-path-id* | **summary**]

**Syntax Description**

| | |
|---|---|
| *type* | (Optional) Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Physical interface or a virtual interface.<br>**Note** Use the **show interfaces**command to see a list of all possible interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **summary** | (Optional) Displays a summary of OOR events. |

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

**Examples**

The following is sample output from the **show rsvp counters oor** command:

```
RP/0/RP0/CPU0:router# show rsvp counters oor

  HundredGigE 0/0/0/3    Rejected
      Path                  24
  HundredGigE 0/0/0/4    Rejected
      Path                  31
  All RSVP Interfaces    Rejected
      Path                  55
```

This table describes the significant fields shown in the display.

***Table 6: show rsvp counters oor Command Field Descriptions***

| Field | Description |
|---|---|
| Path | Number of Path messages received on the interface that were rejected due to oor conditions. |

# show rsvp counters prefix-filtering

To display internal prefix-filtering related RSVP counters, use the **show rsvp counters prefix-filtering** command in XR EXEC mode.

**show rsvp counters prefix-filtering interface** [*type interface-path-id* | **summary**] **access-list** [*aclname*]

## Syntax Description

| | |
|---|---|
| **interface** | Displays RSVP prefix-filtering counters for all interfaces. |
| *type* | (Optional) Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Physical interface or a virtual interface.<br>**Note** Use the **show interfaces** command to see a list of all possible interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **summary** | (Optional) Displays a summary of RSVP prefix-filtering counters on all interfaces. |
| **access-list** | Displays RSVP prefix-filtering counters for the access control list. |
| *aclname* | (Optional) Name of the access control list. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

Counters do not increment if you have not configured an access control list for prefix-filtering.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following is sample output from the **show rsvp counters prefix-filtering** command:

```
RP/0/RP0/CPU0:router# show rsvp counters prefix-filtering interface

  Routed        Fwd        Local        Drop         Def-Drop       Def-Proc      Total
  Path            4                                                                 4
  PathTear        0                                                                 0
```

```
  ResvConfirm     0                                                                0
  Total           4                                                                4


HundredGigE0/0/0/3      Fwd        Local        Drop         Def-Drop       Def-Proc     Total

  Path                        1            0              219              2          222
  PathTear                    0            0               31              0           31
  ResvConfirm                 0            0                0              0            0
  Total                       1            0              219              2          253


HundredGigE0/0/0/3   Fwd        Local        Drop         Def-Drop       Def-Proc       Total

  Path                        0            0                0              1            1
  PathTear                    0            0                0              0            0
  ResvConfirm                 0            0                0              0            0
  Total                       0            0                0              1            1

  ALL RSVP
  Interfaces    Fwd        Local        Drop         Def-Drop       Def-Proc       Total
  Path            4            1            0              219              3          227
  PathTear        0            0            0               31              0           31
  ResvConfirm     0            0            0                0              0            0
  Total           4            1            0              250              3          258
```

The following is sample output from the **show rsvp counters prefix-filtering interface** *type interface-path-id*command:

```
RP/0/RP0/CPU0:router# show rsvp counters prefix-filtering interface  HundredGigE 0/0/0/3


HundredGigE0/0/0/3      Fwd        Local        Drop         Def-Drop       Def-Proc      Total

  Path                        1            0              219              2          222
  PathTear                    0            0               31              0           31
  ResvConfirm                 0            0                0              0            0
  Total                       1            0              250              2          253
```

The following is sample output from the **show rsvp counters prefix-filtering interface summary** command:

```
RP/0/RP0/CPU0:router# show rsvp counters prefix-filtering interface summary

  ALL RSVP
  Interfaces    Fwd        Local        Drop         Def-Drop       Def-Proc       Total
  Path            4            1            0              219              3          227
  PathTear        0            0            0               31              0           31
  ResvConfirm     0            0            0                0              0            0
  Total           4            1            0              250              3          258
```

The following is sample output from the **show rsvp counters prefix-filtering access-list banks**command:

```
RP/0/RP0/CPU0:router# show rsvp counters prefix-filtering access-list banks

  ACL: banks             Forward          Local          Drop           Total
  Path                       0              0              0               0
  PathTear                   0              0              0               0
```

```
ResvConfirm                0              0              0              0
Total                      0              0              0              0
```

This table describes the significant fields shown in the display.

***Table 7: show rsvp counters prefix-filtering interface and summary CommandField Descriptions***

| Field | Description |
|---|---|
| Fwd | Number of messages forwarded to the next router.<br>**Note** The messages are counted against the *routed* interface only because RSVP has no record of what interface the messages will be forwarded to. |
| Local | Number of messages not forwarded (because they are locally destined). |
| Drop | Number of messages dropped. |
| Def-Drop | Number of messages dropped when an access control list match returns an implicit deny. (Results when RSVP is configured to drop implicit deny messages.) |
| Def-Proc | Number of messages processed by RSVP when an access control list match returns an implicit deny. |
| Path | Number of Path messages. |
| PathTear | Number of Path Tear messages. |
| ResvConfirm | Number of ResvConfirm messages. |

# show rsvp fast-reroute

To display RSVP Fast-Reroute (FRR) information, use the **show rsvp fast-reroute** command in EXEC mode.

**show rsvp fast-reroute** [**destination** *IP -address*] [**dst-port** *port*] [**session-type** { **lsp-p2p**}] [**source** *IP-address*] [**src-port** *source-port*] [**summary**]

## Syntax Description

| | |
|---|---|
| **destination** *IP-address* | (Optional) Displays the entries that match the specified address. |
| **dst-port** *port* | (Optional) Displays the port address of the destination router. |
| **session-type** | (Optional) Displays the entries that match the specified session type. |
| **lsp-p2p** | Displays the entries that are used for P2P sessions. |
| **source** *IP-address* | (Optional) Displays the IP address of the source network. |
| **src-port** *source-port* | (Optional) Displays the port number of the source router. |
| **summary** | (Optional) Displays summarized information about the FRR database. |

## Command Default

None

## Command Modes

EXEC

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

This is sample output from the **show rsvp fast-reroute** command:

```
RP/0/RP0/CPU0:router# show rsvp fast-reroute

  Type            Destination         TunID          Source          PSBs          RSBs
  --------      ---------------     -----------    ------------      ---------      -------

  LSP4            10.10.10.10           1          10.20.20.20        Ready          Ready
```

This table describes the significant fields shown in the display.

***Table 8: show rsvp fast-reroute Command Field Descriptions***

| Field | Description |
|---|---|
| Type | Type of session. |
| Destination | Destination address of session. |
| TunID | Tunnel ID number. |
| Source | Source address of session. |
| PSBs | PSB FRR[1] state. |
| RSBs | RSB FRR state. |

[1] Fast reroute.

This is sample output from the **show rsvp fast-reroute summary** command:

```
RP/0/RP0/CPU0:router# show rsvp fast-reroute summary

  States             Total          Ready          Act-Wait           Active
  PSBs                 1              1              0                  0
  RSBs                 1              1              0                  0
```

This table describes the significant fields shown in the display.

***Table 9: show rsvp fast-reroute summary Command Field Descriptions***

| Field | Description |
|---|---|
| States | FRR[2] state. |
| Total | Total number of path and reservation states. |
| Ready | Number of states in FRR ready state. No FRR processing has been done on these states. |
| Act-Wait | Number of states in “Active Wait” FRR state.<br>• For PSBs, this indicates that after FRR the path message has not yet been sent.<br>• For RSBs, this indicates that after FRR, the reservation message has not yet been received. |
| Active | Number of states in “Active” FRR state.<br>• For PSBs, this indicates that after FRR the path message has been sent.<br>• For RSBs, this indicates that after FRR, the reservation message has been received. |

[2] Fast reroute.

# show rsvp graceful-restart

To display the local graceful-restart information for RSVP, use the **show rsvp graceful-restart** command in XR EXEC mode.

**show rsvp graceful-restart** [**neighbors**] [*IP-address*] [**detail**]

## Syntax Description

| | |
|---|---|
| **neighbors** | (Optional) Displays single-line status for each neighbor. If this keyword is not specified, only a multiline table entry is displayed showing local graceful-restart information. |
| *IP-address* | (Optional) Address of the neighbor you are displaying. Displays a specific neighbor with that destination address only. If this keyword is not specified, all neighbors are displayed. |
| **detail** | (Optional) Displays multiline status for each neighbor. If this keyword is not specified, only a single-line table entry is displayed. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

Graceful-restart neighbors are displayed in ascending order of neighbor IP address.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following is sample output from the **show rsvp graceful-restart** command:

```
RP/0/RP0/CPU0:router# show rsvp graceful-restart

  Graceful restart: enabled  Number of global neighbors: 1
  Local MPLS router id: 192.168.55.55
  Restart time: 60 seconds  Recovery time: 120 seconds
  Recovery timer: Not running
  Hello interval: 5000 milliseconds Maximum Hello miss-count: 4
```

This table describes the significant fields shown in the display.

***Table 10: show rsvp graceful-restart Command Field Descriptions***

| Field | Description |
|---|---|
| Graceful restart | Indicates whether graceful restart is configured locally. |
| Number of global neighbors | Number of neighbors identified by a unique router ID. |
| Local MPLS router id | Local router ID used for the MPLS applications. |
| Restart time | Amount of time after a loss in hello messages within which RSVP hello session is reestablished. This setting is manually configurable. |
| Recovery time | Local recovery time advertised to neighbors. This is dynamically computed based on the number of LSPs established and is the time used by neighbors to refresh states in the event of a failure. |
| Recovery timer | Countdown timer which, upon expiry, causes un-refreshed data forwarding states to be deleted (usually beginning with a value that is equivalent to the sum of the restart and recovery times). |
| Hello interval | Interval at which hello messages are sent to neighbors. |
| Maximum hello miss-count | Number of hellos from a neighbor that can be missed before declaring hellos down. |

The following is sample output from the **show rsvp graceful-restart neighbors** command, which displays information about graceful restart neighbors in the router:

```
RP/0/RP0/CPU0:router# show rsvp graceful-restart neighbors

  Neighbor          App  State Recovery       Reason                Since  LostCnt
  --------------- ----- ------ -------- ------------ -------------------- --------
  192.168.77.77 MPLS     UP     DONE          N/A  19/12/2016 17:02:25        0
```

This table describes the significant fields shown in the display.

***Table 11: show rsvp graceful-restart neighbors Command Field Descriptions***

| Field | Description |
|---|---|
| Neighbor | Router ID of a global neighbor. |
| App | Application type of a global neighbor (MPLS ). |
| State | State of the hello session to a global neighbor (up, down, INIT). |
| Recovery | State at which the local node is recovering a global neighbor. |
| Reason | Last reason for which communication has been lost for a global neighbor. If none has occurred, this field is marked as N/A. |
| Since | Time at which the current hello state for a global neighbor has been established. |

| Field | Description |
|---|---|
| LostCnt | Number of times hello communication has been lost with a global neighbor. |

The following is sample output from the **show rsvp graceful-restart neighbors detail** command, which displays detailed information about all graceful restart neighbors:

```
RP/0/RP0/CPU0:router# show rsvp graceful-restart neighbors detail

  Neighbor: 192.168.77.77 Source: 192.168.55.55 (MPLS)
    Hello instance for application MPLS
      Hello State: UP        (for 00:20:52)
      Number of times communications with neighbor lost: 0
      Reason: N/A
    Recovery State: DONE
    Number of Interface neighbors: 1
      address: 192.168.55.0
    Restart time: 120 seconds  Recovery time: 120 seconds
    Restart timer: Not running
    Recovery timer: Not running
    Hello interval: 5000 milliseconds  Maximum allowed missed Hello messages: 4
```

This table describes the significant fields shown in the display.

***Table 12: show rsvp graceful-restart neighbors detail Command Field Descriptions***

| Field | Description |
|---|---|
| Neighbor | Router ID of a global neighbor. |
| Source | Local router ID and application type. |
| Hello State | State of the hello instance for the global neighbor (up, down, or init) and duration of the current state. |
| Number of times communications with neighbor lost | Number of times hello communication has been lost with a global neighbor. |
| Reason | Last reason indicating why communication was lost for a global neighbor. If none has occurred, this field is marked as N/A. |
| Recovery State | State at which the local node is recovering a global neighbor. |
| Number of Interface neighbors | Number of interfaces belonging to a global neighbor. |
| Address | IP address of the interface neighbor. |
| Recovery timer | Remote recovery time for a global neighbor. |
| Hello interval | Interval at which hello messages are sent by the remote global neighbor. |
| Maximum allowed missed Hello messages | Number of hellos that can be missed by the remote global neighbor before declaring hellos down. |

# show rsvp hello instance

To display the RSVP hello instances, use the **show rsvp hello instance** command in XR EXEC mode.

**show rsvp hello instance** [*Hostname* `or` *IP-address*] [**detail**]

**Syntax Description**

| | |
|---|---|
| *Hostname* or *IP-address* | (Optional) Address of the neighbor you are displaying. If this argument is not specified, all neighbors are displayed. |
| **detail** | (Optional) Displays multiline status for each hello instance. If this keyword is not specified, only a single-line table entry is displayed. |

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

**Usage Guidelines**

Hello instances are displayed in ascending order of neighbor IP address.

**Task ID**

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

**Examples**

The following is sample output from the **show rsvp hello instance** command, which displays brief information about all hello instances in the router:

```
RP/0/RP0/CPU0:router# show rsvp hello instance

  Neighbor         Type    State    Interface    LostCnt
  ---------------- ------  -------- ------------ --------
  192.168.77.77    ACTIVE  UP       None         0
```

This table describes the significant fields shown in the display.

***Table 13: show rsvp hello instance Command Field Descriptions***

| Field | Description |
|---|---|
| Neighbor | Router ID of a global neighbor hosting the hello instance. |
| Type | Hello instance type (active or passive). Active type indicates that a node is sending hello requests and passive indicates that a node is sending hello acknowledgements. |

| Field | Description |
|---|---|
| State | State of the hello session to a global neighbor (up, down, or init). |
| Interface | Interface for interface bound hello's used for FRR[3]. Hello instances bound to a global neighbor show Interface as None. Hellos used for FRR are currently not supported. |
| LostCnt | Number of times hello communication has been lost with a global neighbor. |

[3] Fast reroute.

The following is sample output from the **show rsvp hello instance** command, which displays detailed information about all hello instances in the router:

```
RP/0/RP0/CPU0:router# show rsvp hello instance detail

  Neighbor: 192.168.77.77 Source: 192.168.55.55 (MPLS)
    State: UP        (for 00:07:14)
    Type: ACTIVE   (sending requests)
    I/F: None
    Hello interval (msec) (used when ACTIVE)
    Configured: 5000
    Src_instance 0x484b01, Dst_instance 0x4d4247
    Counters:
    Communication with neighbor lost:
      Num of times: 0    Reasons:
        Missed acks:             0
        New Src_Inst received:   0
        New Dst_Inst received:   0
        I/f went down:           0
        Neighbor disabled Hello: 0
    Msgs Received:   93
      Sent:       92
      Suppressed: 87
```

This table describes the significant fields shown in the display.

***Table 14: show rsvp hello instance detail Command Field Descriptions***

| Field | Description |
|---|---|
| Neighbor | Router ID of a global neighbor. |
| Source | Local router ID and application type. |
| State | State of the hello instance for the global neighbor (up, down or init) and duration of the current state. |
| Type | Hello instance type (active or passive). Active type indicates that a node is sending hello requests and passive indicates that a node is sending hello acks. |
| I/F | Interface for interface bound hellos. Hello instances for Graceful restart show interface as None. |

# show rsvp hello instance interface-based

To display the RSVP hello instances on a specific interface, use the **show rsvp hello instance interface-based** command in XR EXEC mode.

**show rsvp hello instance interface-based** [*IP-address*] [**detail**]

**Syntax Description**

| | |
|---|---|
| *IP-address* | (Optional) Address of the neighboring interface. you are displaying. If this argument is not specified, all neighbors are displayed. |
| **detail** | (Optional) Displays detailed information for the specified interface. |

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

**Usage Guidelines**

Hello instances are displayed in ascending order of neighbor IP address.

**Task ID**

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

**Examples**

The following is sample output from the **show rsvp hello instance interface-based** command, which displays detailed information about hello instances on a specific interface:

```
RP/0/RP0/CPU0:router# show rsvp hello instance interface-based 10.10.10.10

  Neighbor         Type    State    Interface    LostCnt
  ---------------- ------  -------- ------------ --------
  10.10.10.10      ACTIVE  UP       None         0
```

This table describes the significant fields shown in the display.

***Table 15: show rsvp hello instance interface-based Command Field Descriptions***

| Field | Description |
|---|---|
| Neighbor | Router ID of a global neighbor hosting the hello instance. |
| Type | Hello instance type (active or passive). Active type indicates that a node is sending hello requests and passive indicates that a node is sending hello acknowledgements. |

| Field | Description |
|---|---|
| State | State of the hello session to a global neighbor (up, down, or init). |
| Interface | Interface for interface bound hellos used for FRR[4]. For hello instances bound to a global neighbor, interface will be displayed as none. |
| LostCnt | Number of times hello communication has been lost with a global neighbor. |

[4] Fast reroute.

# show rsvp neighbor

To display information about RSVP neighbors, use the **show rsvp neighbor** command in XR EXEC mode.

**show rsvp neighbor** [**detail**]

**Syntax Description**

| | |
|---|---|
| **detail** | (Optional) Displays detailed information about RSVP neighbors. |

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

**Usage Guidelines**

Use the **show rsvp interface** command to display various configuration settings such as the list of neighbors and their refresh reduction capabilities.

**Task ID**

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

**Examples**

The following is sample output from the **show rsvp neighbor** command using the **detail** keyword:

```
RP/0/RP0/CPU0:router# show rsvp neighbor detail

  Global Neighbor: 10.10.10.10
  Interface Neighbor: 10.0.0.1
  Interface: HundredGigE0/0/0/3
  Refresh Reduction: "Enabled" or "Disabled".
  Remote epoch: 0xXXXXXXXX
  Out of order messages: 0
  Retransmitted messages: 0
  Interface Neighbor: 172.16.0.1
  Interface: HundredGigE0/0/0/3
  Refresh Reduction: "Enabled" or "Disabled".
  Remote epoch: 0xXXXXXXXX
  Out of order messages: 0
  Retransmitted messages: 0
```

# show rsvp reservation

To display all reservations that RSVP knows about on a router, use the **show rsvp reservation** command in XR EXEC mode.

**show rsvp reservation** [**destination** *IP address*] [**detail**] [**dst-port** *port-num*] [**session-type** { **lsp-p2p** }] [**source** *IP-address*] [**src-port** *port-num*]

## Syntax Description

| | |
|---|---|
| **detail** | (Optional) Displays multiline status for each reservation. If the **detail** keyword is not specified, only a single-line table entry is displayed. |
| **destination** *IP-address* | (Optional) Displays the entries that match the specified address. |
| **dst-port** *port-num* | (Optional) Displays destination port and tunnel ID information. |
| **session-type** | (Optional) Displays the entries that match the specified session type. |
| **lsp-p2p** | Displays the entries that are used for P2P sessions. |
| **source** *IP-address* | (Optional) Displays source address information. |
| **src-port** *port-num* | (Optional) Displays source port and LSP ID information. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

The **show rsvp reservation** command displays information about downstream reservations only (that is, reservations received on this device or created by application program interface (API) calls). Upstream reservations or requests are displayed using the **show rsvp request** command.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following is sample output from the **show rsvp reservation** command:

```
RP/0/RP0/CPU0:router# show rsvp reservation

         Dest Addr DPort      Source Addr SPort Pro   Input IF Sty Serv Rate Burst
 ---------------- ----- ---------------- ----- --- ---------- --- ---- ---- -----
    192.168.40.40  2001   192.168.67.68     2   0       None  SE LOAD     0    1K
    192.168.67.68  2000     10.40.40.40    15   0  HundredGigE 0/0/0/3  SE LOAD     0
```

```
1K
```

The following example displays detailed information about all reservations in the router:

```
RP/0/RP0/CPU0:router# show rsvp reservation detail

  RESV: IPv4-LSP Session addr: 192.168.40.40. TunID: 2001. LSPId: 2.
   Source addr: 192.168.67.68. ExtID: 192.168.67.68.
   Input adjusted interface: None. Input physical interface: None.
   Next hop: 0.0.0.0 (lih: 0x0).
   Style: Shared-Explicit. Service: Controlled-Load.
   Rate: 0 bits/sec. Burst: 1K bytes. Peak: 0 bits/sec.
   MTU min: 40, max: 500 bytes.
   Flags: Local Receiver.
   State expires in 0.000 sec.
   Policy:  Accepted. Policy source(s): MPLS/TE.
   Header info: RSVP TTL=255. IP TTL=255. Flags: 0x0. TOS=0xff.
   Resource:
    Labels: Local downstream: 3.

  RESV: IPv4-LSP Session addr: 192.168.67.68. TunID: 2000. LSPId: 15.
   Source addr: 192.168.40.40. ExtID: 10.10.40.40.
   Input adjusted interface: HundredGigE 0/0/0/3. Input physical interface: HundredGigE
0/0/0/3.
   Next hop: 10.66.67.68 (lih: 0x8DE00002).
   Style: Shared-Explicit. Service: Controlled-Load.
   Rate: 0 bits/sec. Burst: 1K bytes. Peak: 0 bits/sec.
   MTU min: 0, max: 500 bytes.
   Flags: None.
   State expires in 361.184 sec.
   Policy:  Accepted. Policy source(s): MPLS/TE.
   Header info: RSVP TTL=254. IP TTL=254. Flags: 0x1. TOS=0xff.
   Resource:
    Labels: Outgoing downstream: 3.
```

This table describes the significant fields shown in the display.

***Table 16: show rsvp reservation detail Command Field Descriptions***

| Field | Description |
|---|---|
| Input adjusted interface | Interface to reflect the path's outgoing interface. |
| Input physical interface | Interface where the reservation was received. |
| Next hop | Address of the downstream node that sent the reservation to this node. |
| Lih | Logical interface handle sent in the hop object of path returned to us in the reservation to figure out what interface the path was sent on. |
| Flags | Indicates path state, including as Local Repair, Local Sender (LSP[5] ingress node), and others. |
| Policy | Admission control status. |
| Policy source | Entity performing the admission control on the LSP. |
| Header info | RSVP header information as described in RFC 2205. |

[5] Link-state packet

# show rsvp sender

To display all path states that RSVP knows about on this router, use the **show rsvp sender** command in XR EXEC mode mode.

**show rsvp sender** [**destination** *IP-address*] [**detail**] [**dst-port** *port-num*] [**session-type** { **lsp-p2p** }] [**source** *IP-address*] [**src-port** *port-num*]

## Syntax Description

| | |
|---|---|
| **detail** | (Optional) Displays multiline status for each path. If the **detail** keyword is not specified, only a single-line table entry is displayed. |
| **destination** *IP-address* | (Optional) Displays the entries that match the specified address. |
| **dst-port** *port-num* | (Optional) Displays destination port and tunnel ID information. |
| **session-type** | (Optional) Displays the entries that match the specified session type. |
| **lsp-p2p** | Displays the entries that are used for P2P sessions. |
| **source** *IP-address* | (Optional) Displays source address information. |
| **src-port** *port-num* | (Optional) Displays source port and LSP ID information. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

The **show rsvp sender** command displays information about path states.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following is sample output from the **show rsvp sender**command:

```
RP/0/RP0/CPU0:router# show rsvp sender

  Dest Addr        DPort    Source Addr SPort   Pro  Input IF            Rate Burst Prot
  ---------------- ----- ---------------- ----- --- ------------        ---- ----- ----
  10.40.40.40     2001      10.66.67.68     2   0    HundredGigE0/0/0/3    0    1K  Off
  10.66.67.68     2000      10.40.40.40    15   0    None                  0    1K  Off
```

This table describes the significant fields shown in the display.

***Table 17: show rsvp sender Command Field Descriptions***

| Field | Description |
|---|---|
| DProt | Destination port number and tunnel-id. |
| Dest Address | Destination and session address of LSP[6]. |
| SPort | Source port and LSP ID. |
| Source Addr | Address of the ingress node of the LSP. |
| Input IF | Interface on which the Path message was received. |

[6] Link-state packet

The following example displays detailed information about all paths in the system:

```
RP/0/RP0/CPU0:router# show rsvp sender detail

  PATH: IPv4-LSP Session addr: 10.66.67.68. TunID: 1. LSPId: 25.
   Source addr: 10.40.40.40. ExtID: 10.40.40.40.
   Prot: Off. Backup tunnel: None.
   Setup Priority: 7, Reservation Priority: 0
   Rate: 0 bits/sec. Burst: 1K bytes. Peak: 0 bits/sec.
   Min unit: 40 bytes, Max unit: 500 bytes
   Flags: Bidirectional.
   State expires in 370.154 sec.
   Policy:  Accepted. Policy source(s): Default.
   Header info: RSVP TTL=254. IP TTL=254. Flags: 0x1. TOS=0xc0.
   Input interface: HundredGigE 0/0/0/3. Previous hop: 10.40.40.40 (lih: 0x40600001).
   Resource:
    Labels: Outgoing upstream: 3.
   Class-Type: None.
   Explicit Route (Incoming):
       Strict, 10.66.67.68(interface-path-id 5)
       Strict, 10.66.67.68/32
```

This table describes the significant fields shown in the display.

***Table 18: show rsvp sender detail Command Field Descriptions***

| Field | Description |
|---|---|
| Prot | LSP configured as a protected tunnel. |
| Backup tunnel | Name of the backup tunnel assigned to protect this LSP[7]. |
| Flags | Path state, including as local repair, local sender (LSP ingress node), and others. |
| Policy | Admission control status for Path message in the incoming direction. |
| Policy source | Entity doing the admission control, such as COPS or MPLS-TE[8]. |
| Header info | RSVP header information as described in RFC 2205. |

| Field | Description |
|---|---|
| Input interface | Interface on which the path was received. At ingress mode, it is None. |
| Previous hop | Address of the upstream peer who sent us the Path message. May be the interface address or node-id depending on LSP (packet or optical). |
| Lih | Logical interface handle received in the hop object of the path. |
| Output interface | Interface on which the path was forwarded to the downstream neighbor |
| Policy | Admission control status for the path in the outgoing direction. |
| Explicit route | Explicit route specified in the explicit-route object of the Path message. |

[7] Link-state packet
[8] MPLS-Traffic Engineering

# show rsvp session

To list all sessions that RSVP knows about on this router, use the **show rsvp session** command in XR EXEC mode.

**show rsvp session** [**destination** *IP-address*] [**detail**] [**dst-port** *port-num*] [**session-type** { **lsp-p2p** }] [**tunnel-name** *tunnel-name*]

## Syntax Description

| | |
|---|---|
| **detail** | (Optional) Displays multiline status for each path. If the **detail** keyword is not specified, only a single-line table entry is displayed. |
| **destination** *IP-address* | (Optional) Displays the entries that match the specified address. |
| **dst-port** *port-num* | (Optional) Displays destination port and tunnel ID information. |
| **session-type** | (Optional) Displays the entries that match the specified session type. |
| **lsp-p2p** | Displays the entries that are used for P2P sessions. |
| **tunnel-name** *tunnel-name* | (Optional) Displays status for the session matching the specified tunnel-name. |

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

Sessions are displayed in ascending order of destination IP address, destination port, and source IP address.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following is sample output from the **show rsvp session** command:

```
RP/0/RP0/CPU0:router# show rsvp session

  Type    Session Addr  Port  Proto/ExtTunID  PSBs  RSBs  Reqs
  ---- --------------- ----- --------------- ----- ----- -----
  LSP4     10.40.40.40  2001     10.66.67.68     1     1     1
  LSP4     10.66.67.68  2000     10.40.40.40     1     1     0
```

This table describes the significant fields shown in the display.

*Table 19: show rsvp session Command Field Descriptions*

| Field | Description |
|---|---|
| Type | Type of data flow (Traffic-Engineering LSP (LSP4 or IPV4 session). |
| Session Addr | Destination address of the data packets and also tail of the LSP. |
| Port | Destination port or tunnel ID in case of TE tunnels. |
| Proto/ExtTunID | Source address of TE tunnels or protocol as in the case of IPV4 sessions. |
| PSBs | Number of path state blocks for this session. |
| RSBs | Number of reservation state blocks pertaining to incoming or local reservations for this session. |
| Reqs | Number of requests. State data structure representing reservations sent up-stream. |

The following is sample output for the **show rsvp session detail** command:

```
RP/0/RP0/CPU0:router# show rsvp session detail

  SESSION: IPv4-LSP Addr: 10.66.67.68, TunID: 1, ExtID: 10.40.40.40
   PSBs: 1, RSBs: 1, Requests: 0
   LSPId: 1
   Tunnel Name: newhead_t1
    RSVP Path Info:
     InLabel: No intf, No label
     Incoming Address: Unknown
     Explicit Route:
       Strict, 10.66.67.68(interface-path-id 5)
       Strict, 10.66.67.68/32
     Record Route: None
     Tspec: avg rate=0, burst=1K, peak rate=0
    RSVP Resv Info:
     OutLabel: HundredGigE0/0/0/3, 5
     FRR OutLabel: No intf, No label
     Record Route:
       Node-id 10.66.67.68, interface index 5
     Fspec: avg rate=0, burst=1K, peak rate=0
```

This table describes the significant fields shown in the display.

*Table 20: show rsvp session detail Command Field Descriptions*

| Field | Description |
|---|---|
| TunID | Tunnel identifier and the destination port of the LSP[9]. |
| ExtID | Ingress node address of LSP. |
| Tunnel Instance | Source port of the LSP (with the ExtId forming the source parameters). |
| Tunnel Name | Name of the tunnel and LSP. |

| Field | Description |
|---|---|
| InLabel | Incoming interface and label info for the LSP in the upstream direction. At the egress node, using penultimate hop popping at the egress node, (implicit-null) appears as *No Label*. |
| Incoming Address | Address of the ingress interface. |
| Explicit Route | Explicit route specified in the explicit-route object of the Path message. |
| Record Route | Record route object in either the path or reservation message. |
| Tspec | Traffic parameters. |
| OutLabel | Outgoing interface and label sent downstream. |
| FRR OutLabel | For FRR[10], displays the backup tunnel and Merge-point label. |
| Fspec | Flow spec parameters for specified QoS. |

[9] Link-state packet.
[10] Fast reroute.

# signalling dscp (RSVP)

To give all RSVP signaling packets sent out on a specific interface higher priority in the network by marking them with a particular Differentiated Service Code Point (DSCP), use the **signalling dscp** command in RSVP interface configuration submode. To return to the default behavior, use the **no** form of this command.

**signalling dscp** *dscp*
**no signalling** *dscp*

## Syntax Description

| | |
|---|---|
| *dscp* | DSCP priority number. Range is 0 to 63. |

## Command Default

No default behavior or values

## Command Modes

RSVP interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

DSCP marking improves signaling setup and teardown times.

Usually, when a router receives Path messages for a particular state marked with a DSCP value, it sends out Path messages for that state marked with the same DSCP value. This command overrides that DSCP persistence and ensures that all messages sent out a particular interface are marked with a specified DSCP.

Though this command controls RSVP signaling packets, it has no effect on ordinary IP or MPLS data packets traveling along the path created or reserved by this RSVP session.

DSCP persistence operates on a per-state basis, but this command operates on a per-interface basis. So, if some incoming message (for example, multicast Path) with DSCP 10 causes two outgoing messages on interfaces A and B, usually both are sent with DSCP 10. If **signalling dscp 5** is configured for RSVP on interface A, the Path messages being sent out interface A is marked with DSCP 5, but the Path messages being sent out of interface B are marked with DSCP 10.

There is a difference between **signalling dscp 0** and **no signalling dscp** commands. The first command instructs RSVP to explicitly set the DSCP value to 0 on all packets sent out of this interface. The second command removes any override on the packets being sent out of this interface, and allows the DSCP of received packets that created this state to persist on packets forwarded out of this interface.

The RFC specifies a standard mapping from the eight IP precedence values to eight values in the 64-value DSCP space. You can use those special DSCP values to specify IP precedence bits only.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following example shows how to mark all RSVP packets going out on a HundredGigE interface with DSCP value 20.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface HundredGigE 0/0/0/3
RP/0/RP0/CPU0:router(config-rsvp-if)# signalling dscp 20
```

The following example shows how to disable DSCP marking of signaling packets going out on a HundredGigE interface.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface HundredGigE 0/0/0/3
RP/0/RP0/CPU0:router(config-rsvp-if)# no signalling dscp
```

# signalling graceful-restart

To enable or disable RSVP signaling graceful restart, use the **signalling graceful-restart** command in RSVP configuration mode. To return to the default behavior, use the **no** form of this command.

**signalling graceful-restart** [**recovery-time** *time* | **restart-time** *time*]
**no signalling graceful-restart**

## Syntax Description

| | |
|---|---|
| **recovery-time** | (Optional) Configures the recovery time that is advertised in the Restart Cap object in the Hello messages. |
| *time* | Time, in seconds, for the neighbor to wait for the node to recover (replay) existing states after the Hello session is reestablished before initiating TEARs. Range is 0 to 3600. |
| **restart-time** | (Optional) Configures the restart time that is advertised in the Restart Cap object in hello messages. |
| *time* | Time, in seconds, after a control-plane restart that RSVP can start exchanging hello messages. Range is 60 to 3600. Default is 120. |

## Command Default

RSVP signaling graceful restart is disabled.

## Command Modes

RSVP configuration

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

The **signalling graceful-restart** command provides a mechanism that helps minimize the negative effects on MPLS and Optical User Network Interface (O-UNI) traffic for the following types of faults. This is an implementation of the fault handling section of the IETF standard RFC 3473:

**Control-channel-failure**

Disruption of communication channels between 2 nodes when the communication channels are separated from the data channels.

**Node-failure**

Control plane of a node fails, but the node preserves its data forwarding states.

The **signalling graceful-restart** command instigates the exchange of RSVP hello messages between the router and its neighbor nodes. After the hello messages are established with a given neighbor, RSVP can detect these types of faults when they occur.

If no hello messages are received from a neighbor within a certain number of hello intervals, a node assumes that communication with the neighbor has been lost. The node waits the amount of time advertised by the last restart time communicated by the neighbor, before invoking procedures for recovery from communication loss.

The configured restart time is important in case of recovery from failure. The configured value should accurately reflect the amount of time within which, after a control-plane restart, RSVP can start exchanging hello messages.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following example shows how to enable RSVP signalling graceful restart:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp
RP/0/RP0/CPU0:router(config-rsvp)# signalling graceful-restart
```

The following example shows how to set the restart time:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp
RP/0/RP0/CPU0:router(config-rsvp)# signalling graceful-restart restart-time 200
```

The following example shows how to reset the restart time to the default of 120 seconds:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp
RP/0/RP0/CPU0:router(config-rsvp)# no signalling graceful-restart restart-time
```

# signalling hello graceful-restart interface-based

To enable RSVP to accept interface-based hello requests from the neighbor on an interface and send a Hello Acknowledgment to it, use the **signalling hello graceful-restart interface-based** command in RSVP configuration mode. To return to the default behavior, use the **no** form of this command.

**signalling hello graceful-restart interface-based**
**no signalling hello graceful-restart interface-based**

## Syntax Description

This command has no arguments or keywords.

## Command Default

No default behavior or values

## Command Modes

RSVP interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following example shows how to enable interface-based graceful restart:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface tunnel-te 66
RP/0/RP0/CPU0:router(config-rsvp-if)# signalling hello graceful-restart interface based
```

# signalling hello graceful-restart refresh interval

To configure the interval at which RSVP graceful-restart hello messages are sent to each neighbor, use the **signalling hello graceful-restart refresh interval** command in RSVP configuration mode. To return to the default behavior, use the **no** form of this command.

**signalling hello graceful-restart refresh interval** *refresh-interval*
**no signalling hello graceful-restart refresh interval**

## Syntax Description

| | |
|---|---|
| *refresh-interval* | Interval, in milliseconds, at which RSVP graceful-restart hello messages are sent to each neighbor. Range is 3000 to 30000. |

## Command Default

*refresh interval*: 5000

## Command Modes

RSVP configuration

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

The **signalling hello graceful-restart refresh interval** command determines how often hello messages are sent to each neighbor. If the interval is made short, the hello messages are sent more frequently. Although a short interval may help detect failures quickly, it also results in increased network traffic. Optimizations in the RSVP hello mechanism exist to reduce the number of hello messages traveling over the network.

When an RSVP hello message is received, the receiving node acknowledges the hello and restarts its hello timer to the neighbor. By doing this, a hello is transmitted to the neighbor only if a hello is not received before the hello refresh interval has expired.

If two neighboring nodes do not have the same hello interval, the node with the larger hello interval has to acknowledge its neighbor's (more frequent) hellos. For instance, if node A has a hello interval of 5 seconds, and node B has a hello interval of 10 seconds, node B still has to send hello messages every 5 seconds.

The hello backoff mechanism is an optimization that is tailored to minimize the number of hello messages from a neighbor that either does not have graceful restart enabled, or that fails to come back up during the restart interval. The restart interval is provided by the neighbor in the restart cap object.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following example sets the hello graceful-restart refresh interval to 4000 msecs:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp
```

```
RP/0/RP0/CPU0:router(config-rsvp)# signalling hello graceful-restart refresh interval 4000
```

# signalling prefix-filtering access-list

To specify the extended access control list to use for prefix filtering of RSVP Router Alert messages, use the **signalling prefix-filtering access-list** command in RSVP configuration mode. To return to the default behavior, use the **no** form of this command.

**signalling prefix-filtering access-list** *access list name*
**no signalling prefix-filtering access-list** *access list name*

## Syntax Description

| | |
|---|---|
| *access list name* | Extended access-list name as a string (maximum 32 characters). |

## Command Default

No default behavior or values

## Command Modes

RSVP configuration

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

The extended access control list containing the source and destination prefixes used for packet filtering is configured separately.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following example shows how to configure the access control list name banks for prefix-filtering of RSVP Router Alert messages:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp
RP/0/RP0/CPU0:router(config-rsvp)# signalling prefix-filtering access-list banks
```

The following example shows how to disable RSVP prefix-filtering of RSVP Router Alert messages:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp
RP/0/RP0/CPU0:router(config-rsvp)# no signalling prefix-filtering access-list banks
```

# signalling prefix-filtering default-deny-action

To configure RSVP to drop RSVP Router Alert messages when an access control list match returns an implicit deny, use the **signalling prefix-filtering default-deny-action** command in RSVP configuration mode. To return to the default behavior, use the **no** form of this command.

**signalling prefix-filtering default-deny-action drop**
**no signalling prefix-filtering default-deny-action drop**

## Syntax Description

| | |
|---|---|
| **drop** | Specifies when RSVP router alert messages are dropped. |

## Command Default

Performs normal RSVP processing of Path, Path Tear, and ResvConfirm message packets.

## Command Modes

RSVP configuration

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following example shows how to configure RSVP Router Alert messages when an access control list match returns an implicit deny:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp
RP/0/RP0/CPU0:router(config-rsvp)# signalling prefix-filtering default-deny-action drop
```

# signalling rate-limit

To limit the rate of RSVP signaling messages being sent out a particular interface, use the **signalling rate-limit** command in RSVP interface configuration mode. To return to the default behavior, use the **no** form of this command.

**signalling rate-limit**[**rate** *messages*] [**interval** *interval-length*]
**no signalling rate-limit** [**rate** *messages*] [**interval** *interval-length*]

## Syntax Description

| | |
|---|---|
| **rate** *messages* | (Optional) Configures the number of messages sent per scheduling interval. Range is 1 to 500 messages. |
| **interval** *interval-length* | (Optional) Specifies the length, in milliseconds, between scheduling intervals. Range is 250 to 2000. |

## Command Default

*messages*: 100

*interval-length*: 1,000 (1 second)

## Command Modes

RSVP interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

Use the rate-limiting feature with caution. Limiting the rate of RSVP signaling has the advantage of avoiding an overload of the next hop router's input queue, because such overloads would cause the next hop router to drop RSVP messages. However, reliable messaging and rapid retransmit usually enable the router to recover very rapidly from message drops; so rate limiting might not be necessary.

If the rate is set too low, it causes slower convergence times. This command limits all RSVP messages except acknowledgments (ACK) and SRefresh messages. The command does not let you make a router generate messages faster than its inherent limit. (That limit differs among router models.)

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following example shows how to enable rate-limiting:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface HundredGigE 0/0/0/3
RP/0/RP0/CPU0:router(config-rsvp-if)# signalling rate-limit
```

The following example shows how to limit the rate to 50 messages per second:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface HundredGigE 0/0/0/3
RP/0/RP0/CPU0:router(config-rsvp-if)# signalling rate-limit rate 50
```

The following example shows how to set a limit at 40 messages for every 250 milliseconds:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface HundredGigE 0/0/0/3
RP/0/RP0/CPU0:router(config-rsvp-if)# signalling rate-limit rate 40 interval 250
```

The following example shows how to restore the rate to the default of 100 messages per second:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface HundredGigE 0/0/0/3
RP/0/RP0/CPU0:router(config-rsvp-if)# no signalling rate-limit rate
```

The following example shows how to disable rate-limiting:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface HundredGigE 0/0/0/3
RP/0/RP0/CPU0:router(config-rsvp-if)# no signalling rate-limit
```

# signalling refresh interval

To change the frequency with which a router updates the network about the RSVP state of a particular interface, use the **signalling refresh interval** command in RSVP interface configuration mode. To return to the default behavior, use the **no** form of this command.

**signalling refresh interval** *seconds*
**no signalling refresh interval**

## Syntax Description

| | |
|---|---|
| *seconds* | Number of seconds the router waits to update the network about the RSVP state of an interface, in seconds. Range is 10 to 180. Default is 45. |

## Command Default

*seconds*: 45

## Command Modes

RSVP interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

RSVP relies on a soft-state mechanism to maintain state consistency in the face of network losses. That mechanism is based on continuous refresh messages to keep a state current. Each RSVP router is responsible for sending periodic refresh messages to its neighbors.

The router attempts to randomize network traffic and reduce metronomic burstiness by jittering the actual interval between refreshes by as much as 50 percent. As a result, refreshes may not be sent at exactly the interval specified. However, the average rate of refreshes are within the specified refresh interval.

Lengthening the interval reduces the refresh load of RSVP on the network but causes downstream nodes to hold state longer. This reduces the responsiveness of the network to failure scenarios. Shortening the interval improves network responsiveness but expands the messaging load on the network.

The reliable messaging extension, implemented through the **signalling refresh reduction reliable** command, may cause new or changed messages to be temporarily refreshed at a more rapid rate than specified to improve network responsiveness.

The use of reliable messaging with rapid retransmit substantially improves network responsiveness in case of transient message loss; if the refresh interval is changed when using the reliable messaging feature, it is more useful to lengthen the interval than to shorten it.

The summary refresh extension, implemented through the **signalling refresh reduction summary** command, provides a lower-cost mechanism to refresh RSVP state. The router uses the same refresh interval between successive refreshes of a single state when using summary refresh and when using ordinary message-based refresh.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following example shows how to specify a refresh interval of 30 seconds:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface tunnel-te 2
RP/0/RP0/CPU0:router(config-rsvp-if)# signalling refresh interval 30
```

The following example shows how to restore the refresh interval to the default value of 45 seconds:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface tunnel-te 2
RP/0/RP0/CPU0:router(config-rsvp-if)# no signalling refresh interval
```

# signalling refresh missed

To specify the number of successive refresh messages that can be missed before the RSVP deems a state to be expired (resulting in the state to be torn down), use the **signalling refresh missed** command in RSVP interface configuration mode. To return to the default behavior, use the **no** form of this command.

**signalling refresh missed** *number*
**no signalling refresh missed**

## Syntax Description

| | |
|---|---|
| *number* | Number of successive missed refresh messages. Range is 1 to 8. Default is 4. |

## Command Default

*number*: 4

## Command Modes

RSVP interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

Decreasing the missed-message number improves RSVP responsiveness to major failures like router failure or link faults, but decreases the resilience of RSVP resulting in packet drops or temporary network congestion. The latter condition makes RSVP too sensitive.

Increasing the missed-message number increases the resilience of RSVP to such transient packet loss, but decreases the RSVP responsiveness to more intransient network failures such as router failure or link fault.

The default value of 4 provides a balance of resilience and responsiveness factors.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following example shows how to specify a missed refresh limit of six (6) messages:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface tunnel-te 2
RP/0/RP0/CPU0:router(config-rsvp-if)# signalling refresh missed 6
```

The following example shows how to return the missed refresh limit to the default value of four (4):

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface tunnel-te 2
RP/0/RP0/CPU0:router(config-rsvp-if)# no signalling refresh missed
```

# signalling refresh reduction bundle-max-size

To configure the maximum size of a single RSVP bundle message, use the **signalling refresh reduction bundle-max-size** command in RSVP interface configuration mode.

**signalling refresh reduction bundle-max-size** *size*

## Syntax Description

| | |
|---|---|
| *size* | Maximum size, in bytes, of a single RSVP bundle message. Range is 512 to 65000. |

## Command Default

*size*: 4096

## Command Modes

RSVP interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following example shows how to set the maximum bundle size of a single RSVP bundle message to 4000:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface tunnel-te 2
RP/0/RP0/CPU0:router(config-rsvp-if)# signalling refresh reduction bundle-max-size 4000
```

# signalling refresh reduction disable

To disable RSVP refresh reduction on an interface, use the **signalling refresh reduction disable** command in RSVP interface configuration mode. To return to the default behavior, use the **no** form of this command.

**signalling refresh reduction disable**
**no signalling refresh reduction disable**

## Syntax Description

This command has no arguments or keywords.

## Command Default

No default behavior or values

## Command Modes

RSVP interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

The following features of the IETF refresh reduction standard RFC 2961 are enabled with this command:

- Setting the refresh-reduction-capable bit in message headers
- Message-ID usage
- Reliable messaging with rapid retransmit, acknowledgment (ACK), and NACK messages
- Summary refresh extension

Because refresh reduction relies on cooperation of the neighbor, the neighbor must also support the standard. If the router detects that a neighbor is not supporting the refresh reduction standard (either through observing the refresh-reduction-enabled 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 is not used on this link. That information is obtained through use of the **show rsvp interface detail** command.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following example shows how to disable RSVP refresh reduction on an interface:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface tunnel-te 2
RP/0/RP0/CPU0:router(config-rsvp-if)# signalling refresh reduction disable
```

The following example shows how to enable RSVP refresh reduction on the interface:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface tunnel-te 2
RP/0/RP0/CPU0:router(config-rsvp-if)# no signalling refresh reduction disable
```

# signalling refresh reduction reliable

To configure the parameters of reliable messaging, use the **signalling refresh reduction reliable** command in RSVP interface configuration mode. To return to the default behavior, use the **no** form of this command.

**signalling refresh reduction reliable** {**ack-max-size** *bytes* | **ack-hold-time** *milliseconds* | **retransmit-time** *milliseconds* | **summary-refresh**}
**no signalling refresh reduction reliable** {**ack-max-size** *bytes* | **ack-hold-time** *milliseconds* | **retransmit-time** *milliseconds* | **summary-refresh**}

## Syntax Description

| | |
|---|---|
| **ack-max-size** | Specifies the maximum size of the RSVP component within a single acknowledgment message. |
| *bytes* | Number of bytes that define the maximum size of an RSVP component. Range is 20 to 65000. |
| **ack-hold-time** | Specifies the maximum amount of time a router holds an acknowledgment before sending it, in an attempt to bundle several acknowledgments into a single acknowledgment message. |
| *milliseconds* | Number of milliseconds that define the acknowledgment hold time. Range is 100 to 5000. |
| **retransmit-time** | Specifies the amount of time the router initially waits for an acknowledgment message before resending the RSVP message. |
| *milliseconds* | Number of milliseconds that define the retransmit time. Range is 100 to 10000. |
| **summary-refresh** | Enables the use of reliable transmission for RSVP summary refresh messages. |

## Command Default

**ack-max-size** *bytes*: 4096

**ack-hold-time** *milliseconds*: 400 (0.4 seconds)

**retransmit-time** *milliseconds*: 900 (0.9 seconds)

## Command Modes

RSVP interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

For reliable messaging to work properly, configure the retransmit time on the send router (A) and acknowledgment hold time on the peer router (B). (Vice versa for messages in reverse direction.)

The retransmit time must be greater than the acknowledgment hold time, so that the acknowledgment message has time to get back to the sender before the message retransmits. We recommend that the retransmit-time interval be at least twice the acknowledgment hold-time interval. If the retransmit-time value is smaller than the acknowledgment hold-time value, then router A retransmits the message even though router B may have received the message and is waiting for an acknowledgment hold time to time out to send the acknowledgment. This causes unnecessary network traffic.

Reducing the value of **ack-max-size** causes more acknowledgment messages to be issued, with fewer acknowledgments contained within each acknowledgment message. However, reducing the acknowledgment-max-size does not speed up the rate at which acknowledgment messages are issued because their frequency is still controlled by the time values (acknowledgment hold time and retransmit time).

To use reliable messaging for summary refresh messages, use the **rsvp interface** *interface-name* and **signalling refresh reduction summary** commands.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following example shows how to set the maximum acknowledgment message size to 4096 bytes on a HundredGigE interface.

```
RP/0/RP0/CPU0:router(config)# rsvp interface HundredGigE 0/0/0/3
RP/0/RP0/CPU0:router(config-rsvp-if)# signalling refresh reduction reliable ack-max-size
4096
```

The following example shows how to return the maximum acknowledgment message size to the default of 1000 bytes on a HundredGigE interface.

```
RP/0/RP0/CPU0:router(config)# rsvp interface HundredGigE 0/0/0/3
RP/0/RP0/CPU0:router(config-rsvp-if)# no rsvp signalling refresh reduction reliable
```

The following example shows how to set the acknowledgment hold time to 1 second.

```
RP/0/RP0/CPU0:router(config)# rsvp interface HundredGigE 0/0/0/3
RP/0/RP0/CPU0:router(config-rsvp-if)# signalling refresh reduction reliable ack-hold-time
1000
```

The following example shows how to return the acknowledgment hold time to the default of 0.4 second.

```
RP/0/RP0/CPU0:router(config)# rsvp interface HundredGigE 0/0/0/3
RP/0/RP0/CPU0:router(config-rsvp-if)# no signalling refresh reduction reliable ack-hold-time
```

The following example shows how to set the retransmit timer to 2 seconds.

```
RP/0/RP0/CPU0:router(config)# rsvp interface HundredGigE 0/0/0/3
RP/0/RP0/CPU0:router(config-rsvp-if)# signalling refresh reduction reliable retransmit-time
 2000
```

The following example shows how to return the retransmit timer to the default of 0.9 seconds.

```
RP/0/RP0/CPU0:router(config)# rsvp interface HundredGigE 0/0/0/3
```

```
RP/0/RP0/CPU0:router(config-rsvp-if)# no signalling refresh reduction reliable
```

The following example shows how to enable the use of reliable transmission for RSVP summary refresh messages.

```
RP/0/RP0/CPU0:router(config-rsvp-if)# signalling refresh reduction reliable summary-refresh
```

# signalling refresh reduction reliable retransmit-time (RSVP configuration)

To set the global RSVP message retransmission interval for all directly connected neighbors at once or remote neighbors connected through a backup tunnel, use the **signalling refresh reduction reliable retransmit-time** command in the RSVP configuration mode. To remove the configuration, use the **no** form of this command.

**signalling refresh reduction reliable retransmit-time** { **all** | **backup-only** } *milliseconds*
**no signalling refresh reduction reliable retransmit-time**

## Syntax Description

| | |
|---|---|
| {**all**\|**backup-only**} | Specifies that the retransmission interval be applied to all RSVP interfaces or only backup tunnel interfaces. |
| *milliseconds* | Number of milliseconds that defines the retransmission interval.<br>Choose a value between 100 and 30000 milliseconds. |

## Command Default

The default RSVP message retransmission interval is 2100 milliseconds (2.1 seconds).

## Command Modes

RSVP configuration

## Command History

| Release | Modification |
|---|---|
| Release 24.1.1 | This command was introduced. |

## Examples

In this example, an RSVP message retransmission interval of 4000 milliseconds is set for all RSVP interfaces:

```
Router(config)# rsvp
Router(config-rsvp)# signalling refresh reduction reliable retransmit-time all 4000
Router(config-rsvp)# interface hundredGigE 0/0/0/0
Router(config-rsvp-if)# bandwidth 200000
Router(config-rsvp-if)# signalling refresh interval 180
Router(config-rsvp-if)# signalling refresh reduction bundle-max-size 1400
Router(config-rsvp-if)# exit
Router(config-rsvp)# commit
```

In this example, an RSVP message retransmission interval of 5000 milliseconds is set for all backup tunnels during FRR:

```
Router(config)# rsvp
Router(config-rsvp)# signalling refresh reduction reliable retransmit-time backup-only 5000

Router(config-rsvp)# interface hundredGigE 0/0/0/0
Router(config-rsvp-if)# bandwidth 200000
Router(config-rsvp-if)# signalling refresh interval 180
Router(config-rsvp-if)# signalling refresh reduction bundle-max-size 1400
Router(config-rsvp-if)# exit
Router(config-rsvp)# commit
```

# signalling refresh reduction summary

To configure RSVP summary refresh message size on an interface, use the **signalling refresh reduction summary** command in RSVP interface configuration mode. To return to the default behavior, use the **no** form of this command.

**signalling refresh reduction summarymax-size** *bytes*
**no signalling refresh reduction summary max-size** *bytes*

## Syntax Description

| | |
|---|---|
| **max-size** *bytes* | Specifies the maximum size, in bytes, of a single RSVP summary refresh message. Range is 20 to 65000. |

## Command Default

*bytes*: 4096

## Command Modes

RSVP interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

Use the **signalling refresh reduction summary** command to specify the maximum size of the summary refresh messages sent. Message size is verified using the **show rsvp interface detail** command.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following example shows how to change the summary message maximum size on an interface:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface tunnel-te 2
RP/0/RP0/CPU0:router(config-rsvp-if)# signalling refresh reduction summary max-size 6000
```

The following example shows how to return the summary message maximum size to the default value on an interface:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface tunnel-te 2
RP/0/RP0/CPU0:router(config-rsvp-if)# no signalling refresh reduction summary max-size 6000
```

# window-size (RSVP)

To specify the maximum number of RSVP authenticated messages that can be received out of sequence, use the **window-size** command in RSVP authentication configuration mode, RSVP interface authentication configuration mode, or RSVP neighbor authentication configuration mode. To disable the window size, use the **no** form of this command.

**window-size** *N*
**no window-size**

## Syntax Description

*N* Size of the window to restrict out-of-sequence messages. Range is 1 to 64. Default is 1. All out-of-sequence messages are dropped.

## Command Default

*N*: 1

## Command Modes

RSVP authentication configuration

RSVP interface authentication configuration

RSVP neighbor authentication configuration

## Command History

| Release | Modification |
|---|---|
| Release 7.0.12 | This command was introduced. |

## Usage Guidelines

Use the **window-size** command to specify the maximum number of authenticated messages that are received out of sequence. 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 out-of-order or out-of-sequence 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 adjust the burst size such that RSVP does not discard such reordered bursts. RSVP checks for duplicate messages within these bursts.

## Task ID

| Task ID | Operations |
|---|---|
| mpls-te | read, write |

## Examples

The following example shows how to configure the size of the window to 33 in RSVP neighbor authentication configuration mode:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp neighbor 10.0.0.1 authentication
RP/0/RP0/CPU0:router(config-rsvp-nbor-auth)# window-size 33
```

The following example shows how to configure the size of the window to 33 in RSVP authentication configuration mode:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp authentication
RP/0/RP0/CPU0:router(config-rsvp-auth)# window-size 33
```

The following example shows how to configure the size of the window to 33 in RSVP interface authentication configuration mode by using the **rsvp interface** command:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rsvp interface HundredGigE 0/0/0/3
RP/0/RP0/CPU0:router(config-rsvp-if)# authentication
RP/0/RP0/CPU0:router(config-rsvp-if-auth)# window-size 33
```