



# Low Latency Queueing (LLQ) for IPSec Encryption Engines

## Feature History

Release	Modification
12.2(13)T	This feature was introduced.
12.2(14)S	This feature was integrated into Cisco IOS Release 12.2(14)S.

This feature module describes the Low Latency Queueing (LLQ) for IPSec encryption engines feature in Cisco IOS Release 12.2(13)T and 12.2(14)S. It includes the following sections:

- [Feature Overview, page 1](#)
- [Supported Platforms, page 3](#)
- [Supported Standards, MIBs, and RFCs, page 4](#)
- [Prerequisites, page 4](#)
- [Configuration Tasks, page 5](#)
- [Monitoring and Maintaining LLQ for IPSec Encryption Engines, page 8](#)
- [Configuration Examples, page 8](#)
- [Command Reference, page 9](#)
- [Glossary, page 10](#)

## Feature Overview

Low Latency Queueing (LLQ) for IPSec encryption engines helps reduce packet latency by introducing the concept of queueing before crypto engines. Prior to this, the crypto processing engine gave data traffic and voice traffic equal status. Administrators now designate voice traffic as priority. Data packets arriving at a router interface are directed into a data packet inbound queue for crypto engine processing. This queue is called the best effort queue. Voice packets arriving on a router interface are directed into a priority packet inbound queue for crypto engine processing. This queue is called the priority queue. The crypto engine undertakes packet processing in a favorable ratio for voice packets. Voice packets are guaranteed a minimum processing bandwidth on the crypto engine.

## Benefits

The Low Latency Queueing (LLQ) for IPSec encryption engines feature guarantees a certain level of crypto engine processing time for priority designated traffic.

**Note**

---

On the Cisco 2600 platform, with the exception of the Cisco 2691 router, the CPU utilization maximizes out before the crypto engine becomes congested, so latency is not improved.

---

**Better Voice Performance**

Voice packets can be identified as priority, allowing the crypto engine to guarantee a certain percentage of processing bandwidth. This feature impacts the end user experience by assuring voice quality if voice traffic is directed onto a congested network.

**Improved Latency and Jitters**

Predictability is a critical component of network performance. The Low Latency Queueing (LLQ) for IPSec encryption engines feature delivers network traffic predictability relating to VPN. With this feature disabled, an end user employing an IP phone over VPN might experience jitter or latency, both symptoms of overall network latency and congestion. With this feature enabled, these undesirable characteristics are dissipated.

## Restrictions

- No per-tunnel QoS policy. An interface QoS policy represents all tunnels.
- Assume the same IP precedence/DSCP marking for inbound and outbound voice packets.
- Assume the IP precedence/DSCP marking for voice packets are done at the source.
- Limited match criteria for voice traffic in the interface QoS policy.
- Assume call admission control is enforced within the enterprise.
- No strict error checking when aggregate policy's bandwidth exceeds crypto engine bandwidth. Only a warning is displayed but configuration is allowed.
- Assume voice packets are either all encrypted or unencrypted.

## Related Features and Technologies

- CBWFQ
- Priority Queueing
- Weighted Fair Queueing

## Related Documents

- *Quality of Service Solutions Command Reference*, Cisco IOS Release 12.2
- *Class-Based Weighted Fair Queueing* feature module, Cisco IOS Release 12.1
- *IP RTP Priority* feature module, Cisco IOS Release 12.0

## Supported Platforms

### 12.2(14)S and higher

The LLQ for IPSec encryption engines feature is supported on the following platform:

- Cisco 7200 series

### 12.2(13)T

The LLQ for IPSec encryption engines feature is supported on all platforms using Cisco IOS Release 12.2(13)T or later, including:

- Cisco 2600 series
- Cisco 3600 series
- Cisco 7100 series
- Cisco 7200 series

## Determining Platform Support Through Cisco Feature Navigator

Cisco IOS software is packaged in feature sets that are supported on specific platforms. To get updated information regarding platform support for this feature, access Cisco Feature Navigator. Cisco Feature Navigator dynamically updates the list of supported platforms as new platform support is added for the feature.

Cisco Feature Navigator is a web-based tool that enables you to quickly determine which Cisco IOS software images support a specific set of features and which features are supported in a specific Cisco IOS image. You can search by feature or release. Under the release section, you can compare releases side-by-side to display both the features unique to each software release and the features in common.

To access Cisco Feature Navigator, you must have an account on Cisco.com. If you have forgotten or lost your account information, send a blank e-mail to [cco-locksmith@cisco.com](mailto:cco-locksmith@cisco.com). An automatic check will verify that your e-mail address is registered with Cisco.com. If the check is successful, account details with a new random password will be e-mailed to you. Qualified users can establish an account on Cisco.com by following the directions found at this URL:

<http://www.cisco.com/register>

Cisco Feature Navigator is updated regularly when major Cisco IOS software releases and technology releases occur. For the most current information, go to the Cisco Feature Navigator home page at the following URL:

<http://www.cisco.com/go/fn>

## Availability of Cisco IOS Software Images

Platform support for particular Cisco IOS software releases is dependent on the availability of the software images for those platforms. Software images for some platforms may be deferred, delayed, or changed without prior notice. For updated information about platform support and availability of software images for each Cisco IOS software release, refer to the online release notes or, if supported, Cisco Feature Navigator.

## Supported Standards, MIBs, and RFCs

### Standards

- No new or modified standards are supported by this feature.

### MIBs

- No new or modified standards are supported by this feature.

To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL:

<http://tools.cisco.com/ITDIT/MIBS/servlet/index>

If Cisco MIB Locator does not support the MIB information that you need, you can also obtain a list of supported MIBs and download MIBs from the Cisco MIBs page at the following URL:

<http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml>

To access Cisco MIB Locator, you must have an account on Cisco.com. If you have forgotten or lost your account information, send a blank e-mail to [cco-locksmith@cisco.com](mailto:cco-locksmith@cisco.com). An automatic check will verify that your e-mail address is registered with Cisco.com. If the check is successful, account details with a new random password will be e-mailed to you. Qualified users can establish an account on Cisco.com by following the directions found at this URL:

<http://www.cisco.com/register>

### RFCs

- No new or modified RFCs are supported by this feature.

## Prerequisites

To use this feature, you should be familiar with the following:

- Access control lists
- Bandwidth management
- CBWFQ

# Configuration Tasks

To configure LLQ for IPSec encryption engines, perform the tasks described in the following section.



## Note

See the [Quality of Service Solutions Command Reference](#), Cisco IOS Release 12.2, to learn more about configuring server policies on interfaces.

- [Defining Class Maps](#) (required)
- [Configuring Class Policy in the Policy Map](#) (required)
- [Configuring Class Policy for a Priority Queue](#) (required)
- [Configuring Class Policy Using a Specified Bandwidth](#) (optional)
- [Configuring the Class-Default Class Policy](#) (optional)
- [Attaching the Service Policy](#) (required)
- [Verifying Configuration of Policy Maps and Their Classes](#) (optional)

## Defining Class Maps

To create a class map containing match criteria against which a packet is checked to determine if it belongs to a class, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# <b>class-map</b> class-map-name	Specifies the name of the class map to be created.
Step 2	Router(config-cmap)# <b>match access-group</b> {access-group   name access-group-name}	Specifies the name of the access control list (ACL) against whose contents packets are checked to determine if they belong to the class.
	or	
	Router(config-cmap)# <b>match</b> <b>input-interface</b> interface-name	Specifies the name of the input interface used as a match criterion against which packets are checked to determine if they belong to the class.
	or	
	Router(config-cmap)# <b>match protocol</b> protocol	Specifies the name of the protocol used as a match criterion against which packets are checked to determine if they belong to the class.

## Configuring Class Policy in the Policy Map

To configure a policy map and create class policies that make up the service policy, begin with the **policy-map** command to specify the policy map name. Then use one or more of the following commands to configure the policy for a standard class or the default class:

- **priority**
- **bandwidth**
- **queue-limit** or **random-detect**

- **fair-queue** (for class-default class only)

For each class that you define, you can use one or more of the commands listed to configure the class policy. For example, you might specify bandwidth for one class and both bandwidth and queue limit for another class.

The default class of the policy map (commonly known as the class-default class) is the class to which traffic is directed if that traffic does not satisfy the match criteria of the other classes defined in the policy map.

You can configure class policies for as many classes as are defined on the router, up to the maximum of 64. However, the total amount of bandwidth allocated for all classes in a policy map must not exceed the minimum committed information rate (CIR) configured for the virtual circuit (VC) minus any bandwidth reserved by the **frame-relay voice bandwidth** and **frame-relay ip rtp priority** commands. If the minimum CIR is not configured, the bandwidth defaults to one half of the CIR. If all of the bandwidth is not allocated, the remaining bandwidth is allocated proportionally among the classes on the basis of their configured bandwidth.

To configure class policies in a policy map, perform the tasks described in the following sections. The task in the first section is required; the tasks in the remaining sections are optional.

## Configuring Class Policy for a Priority Queue

To configure a policy map and give priority to a class within the policy map, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# <b>policy-map</b> policy-map	Specifies the name of the policy map to be created or modified.
Step 2	Router(config-cmap)# <b>class</b> class-name	Specifies the name of a class to be created and included in the service policy.
Step 3	Router(config-pmap-c)# <b>priority</b> bandwidth-kbps	Creates a strict priority class and specifies the amount of bandwidth, in kbps, to be assigned to the class.

## Configuring Class Policy Using a Specified Bandwidth

To configure a policy map and create class policies that make up the service policy, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# <b>policy-map</b> policy-map	Specifies the name of the policy map to be created or modified.
Step 2	Router(config-cmap)# <b>class</b> class-name	Specifies the name of a class to be created and included in the service policy.
Step 3	Router(config-pmap-c)# <b>bandwidth</b> bandwidth-kbps	Specifies the amount of bandwidth to be assigned to the class, in kbps, or as a percentage of the available bandwidth. Bandwidth must be specified in kbps or as a percentage consistently across classes. (Bandwidth of the priority queue must be specified in kbps.)

To configure more than one class in the same policy map, repeat [Step 2](#) and [Step 3](#).

## Configuring the Class-Default Class Policy

The class-default class is used to classify traffic that does not fall into one of the defined classes. Even though the class-default class is predefined when you create the policy map, you still have to configure it. If a default class is not configured, then traffic that does not match any of the configured classes is given best-effort treatment, which means that the network will deliver the traffic if it can, without any assurance of reliability, delay prevention, or throughput.

To configure a policy map and the class-default class, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# <b>policy-map</b> policy-map	Specifies the name of the policy map to be created or modified.
Step 2	Router(config-cmap)# <b>class class-default</b> <i>default-class-name</i>	Specifies the default class so that you can configure or modify its policy.
Step 3	Router(config-pmap-c)# <b>bandwidth</b> bandwidth-kbps	Specifies the amount of bandwidth, in kbps, to be assigned to the class.
	or  Router(config-pmap-c)# <b>fair-queue</b> [ <i>number-of-dynamic-queues</i> ]	

## Attaching the Service Policy

To attach a service policy to the output interface and enable LLQ for IPSec encryption engines, use the following command in map-class configuration mode:

	Command	Purpose
Step 1	Router(config)# <b>interface</b> type number	Specifies the interface using the LLQ for IPSec encryption engines.
Step 2	Router(config-if)# <b>service-policy output</b> policy-map	Attaches the specified service policy map to the output interface and enables LLQ for IPSec encryption engines.

## Verifying Configuration of Policy Maps and Their Classes

To display the contents of a specific policy map or all policy maps configured on an interface, use the following commands in EXEC mode, as needed:

	Command	Purpose
Step 1	Router# <b>show frame-relay pvc dlci</b>	Displays statistics about the PVC and the configuration of classes for the policy map on the specified data-link connection identifier (DLCI).

	Command	Purpose
Step 2	Router# <b>show policy-map interface</b> <i>interface-name</i>	When LLQ is configured, displays the configuration of classes for all policy maps.
Step 3	Router# <b>show policy-map interface</b> <i>interface-name dlci dlci</i>	When LLQ is configured, displays the configuration of classes for the policy map on the specified DLCI.

## Monitoring and Maintaining LLQ for IPsec Encryption Engines

To monitor and maintain LLQ for IPsec encryption engines, use the following command in EXEC mode:

	Command	Purpose
Step 1	Router# <b>show crypto eng qos</b>	Displays quality of service queuing statistics for LLQ for IPsec encryption engines.

For a more detailed list of commands that can be used to monitor LLQ for IPsec encryption engines, see the section [“Verifying Configuration of Policy Maps and Their Classes”](#)

## Configuration Examples

This section provides the following configuration example:

- [LLQ for IPsec Encryption Engines Example](#)

### LLQ for IPsec Encryption Engines Example

In the following example, a strict priority queue with a guaranteed allowed bandwidth of 50 kbps is reserved for traffic that is sent from the source address 10.10.10.10 to the destination address 10.10.10.20, in the range of ports 16384 through 20000 and 53000 through 56000.

First, the following commands configure access list 102 to match the desired voice traffic:

```
Router(config)# access-list 102 permit udp host 10.10.10.10 host 10.10.10.20 range 16384
20000
Router(config)# access-list 102 permit udp host 10.10.10.10 host 10.10.10.20 range 53000
56000
```

Next, the class map voice is defined, and the policy map called policy1 is created; a strict priority queue for the class voice is reserved, a bandwidth of 20 kbps is configured for the class bar, and the default class is configured for WFQ. The service-policy command then attaches the policy map to the fas0/0.

```
Router(config)# class-map voice
Router(config-cmap)# match access-group 102
Router(config)# policy-map policy1
Router(config-pmap)# class voice
Router(config-pmap-c)# priority 50
Router(config-pmap)# class bar
Router(config-pmap-c)# bandwidth 20
Router(config-pmap)# class class-default
Router(config-pmap-c)# fair-queue
Router(config)# interface fas0/0
```

```
Router(config-if)# service-policy output policy1
```

## Command Reference

See the following sections for configuration tasks for the LLQ for IPsec Encryption Engines feature. Each task in the list is identified as either required or optional.

- [show crypto eng qos](#) (optional)

## show crypto eng qos

To monitor and maintain LLQ for IPsec encryption engines, use the **show crypto eng qos** command in global configuration mode.

[show crypto eng qos](#)

<b>Syntax Description</b>	<b>show crypto eng qos</b> Displays whether QoS is enabled on LLQ for IPsec encryption engines.
---------------------------	---

<b>Defaults</b>	No default behavior or values.
-----------------	--------------------------------

<b>Command Modes</b>	Privileged EXEC
----------------------	-----------------

Command History	Release	Modification
	12.2(13)T	This command was introduced in Cisco IOS Release 12.2(13)T.
	12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.

<b>Usage Guidelines</b>	Use the <b>show crypto eng qos</b> command to determine if QoS is enabled on LLQ for IPsec encryption engines.
-------------------------	--

<b>Examples</b>	The following example shows how to determine if LLQ for IPsec encryption engines is enabled:
-----------------	--

```
Router# show crypto eng qos
crypto engine name: Multi-ISA Using VAM2
  crypto engine type: hardware
    slot: 5
    queuing: enabled
  visible bandwidth: 30000 kbps
    llq size: 0
  default queue size/max: 0/64
  interface table size: 32

FastEthernet0/0 (3), iftype 1, ctable size 16, input filter:ip
precedence 5
```

```
class voice (1/3), match ip precedence 5
    bandwidth 500 kbps, max token 100000
    IN match pkt/byte 0/0, police drop 0
    OUT match pkt/byte 0/0, police drop 0

class default, match pkt/byte 0/0, qdrop 0
crypto engine bandwidth:total 30000 kbps, allocated 500 kbps
```

## Glossary

**IKE**—Internet Key Exchange. IKE establishes a shared security policy and authenticates keys for services (such as IPSec). Before any IPSec traffic can be passed, each router/firewall/host must verify the identity of its peer. This can be done by manually entering preshared keys into both hosts or by a CA service.

**IPSec**—IP Security. A framework of open standards that provides data confidentiality, data integrity, and data authentication between participating peers. IPSec provides these security services at the IP layer. IPSec uses IKE to handle the negotiation of protocols and algorithms based on local policy and to generate the encryption and authentication keys to be used by IPSec. IPSec can protect one or more data flows between a pair of hosts, between a pair of security gateways, or between a security gateway and a host.