

Implementing Cisco Quality of Service

Volume 1

Version 2.3

Student Guide

Text Part Number: 97-2810-02



Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV
Amsterdam, The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at www.cisco.com/go/offices.

 CCDE, CCENT, CCSI, Cisco Eos, Cisco HealthPresence, Cisco IronPort, the Cisco logo, Cisco Lumin, Cisco Nexus, Cisco Nurse Connect, Cisco StackPower, Cisco StadiumVision, Cisco TelePresence, Cisco Unified Computing System, Cisco WebEx, DCE, Flip Channels, Flip for Good, Flip Mino, Flip Video, Flip Video (Design), Flipshare (Design), Flip Ultra, and Welcome to the Human Network are trademarks; Changing the Way We Work, Live, Play, and Learn, Cisco Store, and Flip Gift Card are service marks; and Access Registrar, Aironet, AsyncOS, Bringing the Meeting To You, Catalyst, CCDA, CCDP, CCIE, CCIP, CCNA, CCNP, CCSP, CCVP, Cisco, the Cisco Certified Internetwork Expert logo, Cisco IOS, Cisco Press, Cisco Systems, Cisco Systems Capital, the Cisco Systems logo, Cisco Unity, Collaboration Without Limitation, EtherFast, EtherSwitch, Event Center, Fast Step, Follow Me Browsing, FormShare, GigaDrive, HomeLink, Internet Quotient, IOS, iPhone, iQuick Study, IronPort, the IronPort logo, LightStream, Linksys, MediaTone, MeetingPlace, MeetingPlace Chime Sound, MGX, Networkers, Networking Academy, Network Registrar, PCNow, PIX, PowerPanels, ProConnect, ScriptShare, SenderBase, SMARTnet, Spectrum Expert, StackWise, The Fastest Way to Increase Your Internet Quotient, TransPath, WebEx, and the WebEx logo are registered trademarks of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries.

All other trademarks mentioned in this document or website are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (0907R)

DISCLAIMER WARRANTY: THIS CONTENT IS BEING PROVIDED "AS IS." CISCO MAKES AND YOU RECEIVE NO WARRANTIES IN CONNECTION WITH THE CONTENT PROVIDED HEREUNDER, EXPRESS, IMPLIED, STATUTORY OR IN ANY OTHER PROVISION OF THIS CONTENT OR COMMUNICATION BETWEEN CISCO AND YOU. CISCO SPECIFICALLY DISCLAIMS ALL IMPLIED WARRANTIES, INCLUDING WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT AND FITNESS FOR A PARTICULAR PURPOSE, OR ARISING FROM A COURSE OF DEALING, USAGE OR TRADE PRACTICE. This learning product may contain early release content, and while Cisco believes it to be accurate, it falls subject to the disclaimer above.



*Students, this letter describes important
course evaluation access information!*

Welcome to Cisco Systems Learning. Through the Cisco Learning Partner Program, Cisco Systems is committed to bringing you the highest-quality training in the industry. Cisco learning products are designed to advance your professional goals and give you the expertise you need to build and maintain strategic networks.

Cisco relies on customer feedback to guide business decisions; therefore, your valuable input will help shape future Cisco course curricula, products, and training offerings. We would appreciate a few minutes of your time to complete a brief Cisco online course evaluation of your instructor and the course materials in this student kit. On the final day of class, your instructor will provide you with a URL directing you to a short post-course evaluation. If there is no Internet access in the classroom, please complete the evaluation within the next 48 hours or as soon as you can access the web.

On behalf of Cisco, thank you for choosing Cisco Learning Partners for your Internet technology training.

Sincerely,

Cisco Systems Learning

Table of Contents

Volume 1

<i>Course Introduction</i>	<i>1</i>
Overview	1
Learner Skills and Knowledge	2
Course Goal and Objectives	3
Course Flow	4
Additional References	5
Cisco Glossary of Terms	5
Your Training Curriculum	6
Your Training Curriculum	10
<i>Introduction to QoS</i>	<i>1-1</i>
Overview	1-1
Module Objectives	1-1
<i>Understanding the Need for QoS</i>	<i>1-3</i>
Overview	1-3
Objectives	1-3
Converged Networks	1-4
Converged Networks Quality Issues	1-6
Available Bandwidth	1-8
End-to-End Delay	1-10
Example: Effects of Delay	1-10
Packet Loss	1-14
Summary	1-17
<i>Understanding QoS</i>	<i>1-19</i>
Overview	1-19
Objectives	1-19
QoS Defined	1-20
QoS for Converged Networks	1-21
Example: Three Steps to Implementing QoS on a Network	1-21
QoS Requirements	1-22
QoS Traffic Classes	1-26
Example: Traffic Classification	1-26
QoS Policy	1-27
Example: Defining QoS Policies	1-27
Summary	1-30
<i>Implementing QoS</i>	<i>1-31</i>
Overview	1-31
Objectives	1-31
Methods for Implementing QoS Policy	1-32
Legacy CLI	1-34
Modular QoS CLI	1-35
AutoQoS VoIP and Enterprise	1-36
QoS Implementation Methods Compared	1-38
QoS Policy Manager	1-39
Network MIBs for Monitoring QoS	1-41
MIBs for Managing QoS	1-42
Summary	1-45
Module Summary	1-47
References	1-48
Module Self-Check	1-49
Module Self-Check Answer Key	1-51

<i>The Building Blocks of QoS</i>	2-1
Overview	2-1
Module Objectives	2-1
Identifying Models for Implementing QoS	2-3
Overview	2-3
Objectives	2-3
QoS Models	2-4
Best-Effort Model	2-5
IntServ Model	2-7
DiffServ Model	2-12
Summary	2-14
Understanding the Integrated Services Model	2-15
Overview	2-15
Objectives	2-15
Integrated Services Model	2-16
Components of RSVP	2-20
RSVP Interface Bandwidth Queuing	2-21
Enabling RSVP on an Interface	2-22
IntServ and DiffServ Integration	2-23
Summary	2-24
Understanding the Differentiated Services Model	2-25
Overview	2-25
Objectives	2-25
Differentiated Services Model	2-26
DSCP Encoding	2-28
Per-Hop Behaviors	2-31
Backward Compatibility Using the Class Selector	2-36
Summary	2-37
Identifying QoS Mechanisms	2-39
Overview	2-39
Objectives	2-39
QoS Mechanisms	2-40
Classification	2-41
Marking	2-42
Congestion Management	2-43
Congestion Avoidance	2-44
Policing and Shaping	2-45
Link Efficiency Mechanisms	2-47
Link Fragmentation and Interleaving	2-48
Applying QoS to Input and Output Interfaces	2-49
Summary	2-50
Module Summary	2-51
References	2-51
Module Self-Check	2-53
Module Self-Check Answer Key	2-55

<i>Introduction to MQC and Cisco AutoQoS</i>	<i>3-1</i>
Overview	3-1
Module Objectives	3-1
Introducing MQC	3-3
Overview	3-3
Objectives	3-3
Modular QoS CLI	3-4
Example: Advantages of Using MQC	3-4
Modular QoS CLI Components	3-5
Example: Configuring MQC	3-5
Class Maps	3-6
Configuring and Monitoring Class Maps	3-8
Example: Class Map Configuration	3-8
Example: Using the match Command	3-11
Example: Nested Traffic Class to Combine match-any and match-all Characteristics in One Traffic Class	3-11
Policy Maps	3-15
Configuring and Monitoring Policy Maps	3-16
Example: Policy Map Example	3-19
Example: Hierarchical Policy Maps	3-21
Example: Hierarchical Policy-Map Configuration	3-22
Service Policy	3-27
Attaching Service Policies to Interfaces	3-28
Example: Complete MQC Configuration	3-29
Summary	3-30
Introducing Cisco AutoQoS VoIP	3-31
Overview	3-31
Objectives	3-31
Cisco AutoQoS VoIP	3-32
Cisco AutoQoS VoIP Router Platforms	3-36
Cisco AutoQoS VoIP Switch Platforms	3-37
Configuring Cisco AutoQoS VoIP	3-39
Example: Configuring the Cisco AutoQoS VoIP Feature on a High-Speed Serial Interface	3-42
Example: Using the Port-Specific Cisco AutoQoS Macro	3-46
Monitoring Cisco AutoQoS VoIP	3-48
Example: show auto qos and show auto qos interface	3-49
Automation with Cisco AutoQoS	3-57
Summary	3-58
Introducing Cisco AutoQoS for the Enterprise	3-59
Overview	3-59
Objectives	3-59
Cisco AutoQoS for the Enterprise	3-60
Cisco AutoQoS for the Enterprise Router Platforms	3-63
Configuring Cisco AutoQoS for the Enterprise	3-65
Example: Configuring the Cisco AutoQoS Discovery Feature on a High-Speed Serial Interface	3-68
Example: Configuring the Cisco AutoQoS for the Enterprise Feature on a High-Speed Serial Interface	3-68
Monitoring Cisco AutoQoS for the Enterprise	3-69
Example: show auto qos and show auto qos interface	3-70
Summary	3-73
Module Summary	3-75
References	3-76
Module Self-Check	3-77
Module Assessment Answer Key	3-78

Classification and Marking	4-1
Overview	4-1
Module Objectives	4-1
Understanding Classification and Marking	4-3
Overview	4-3
Objectives	4-3
Classification	4-4
Marking	4-5
Classification and Marking at the Data-Link Layer	4-6
Classification and Marking at the Network Layer	4-11
Mapping CoS to Network Layer QoS	4-12
QoS Service Class Defined	4-14
Example: Defining QoS Service Classes	4-15
Implementing a QoS Policy Using a QoS Service Class	4-16
Trust Boundaries	4-24
Summary	4-26
Using MQC for Classification	4-27
Overview	4-27
Objectives	4-27
MQC Classification Options	4-28
Configuring Classification with MQC	4-31
Monitoring Class Maps	4-44
Summary	4-45
Using MQC for Class-Based Marking	4-47
Overview	4-47
Objectives	4-47
Class-Based Marking Overview	4-48
MQC Marking Options	4-49
Configuring Class-Based Marking	4-51
Monitoring Class-Based Marking	4-61
Summary	4-64
Using NBAR for Classification	4-65
Overview	4-65
Objectives	4-65
Network-Based Application Recognition	4-66
NBAR Application Support	4-70
Packet Description Language Module	4-77
NBAR and Custom Protocols	4-79
Protocol Discovery	4-80
Configuring and Monitoring NBAR	4-83
Summary	4-107
Configuring QoS Preclassify	4-109
Overview	4-109
Objectives	4-109
QoS Issues with VPNs	4-110
ToS Byte Preservation	4-111
QoS Preclassify	4-113
Configuring QoS Preclassify	4-116
Monitoring QoS Preclassify	4-118
Summary	4-122

Configuring QoS Policy Propagation via BGP	4-123
Overview	4-123
Objectives	4-123
QoS Policy Propagation via BGP	4-124
QoS and BGP Interaction	4-126
Cisco Express Forwarding	4-127
QPPB Configuration Tasks	4-131
Configuring QPPB	4-132
Example: Configuration	4-136
Example: Configuring QPPB	4-138
Summary	4-145
Configuring LAN Classification and Marking	4-147
Overview	4-147
Objectives	4-147
LAN-Based Classification and Marking	4-149
QoS Trust Boundaries	4-150
LAN Classification and Marking Platforms	4-153
Example: Configure Trust Settings on the Cisco Catalyst 2950 Series Switches	4-162
Example: QoS Mechanisms on the Cisco Catalyst 2950 Switch	4-163
Example: QoS Mechanisms on Cisco Catalyst 2960 Series Switches	4-164
Configuring LAN-Based Classification and Marking	4-175
Monitoring LAN-Based Classification and Marking	4-193
Summary	4-197
Understanding QoS in the Life of a Packet	4-199
Overview	4-199
Objectives	4-199
QoS and Packets	4-200
Life of a High-Priority VoIP Packet	4-201
Life of a Low-Priority FTP Packet	4-209
Summary	4-218
Module Summary	4-219
References	4-220
Module Self-Check	4-223
Module Self-Check Answer Key	4-225

Course Introduction

Overview

Implementing Cisco Quality of Service (QoS) v2.3 provides learners with in-depth knowledge of quality of service (QoS) requirements; conceptual models such as Best-Effort, Integrated Services (IntServ), and Differentiated Services (DiffServ); and the implementation of QoS on Cisco IOS platforms.

The curriculum covers the theory of QoS, design issues, and configuration of various QoS mechanisms to facilitate the creation of effective QoS administrative policies. Case studies and lab activities included in the course help learners apply the concepts that are mastered in individual modules to real-life scenarios.

The course also gives learners design and usage rules for various advanced QoS features and for the integration of QoS with underlying Layer 2 QoS mechanisms, allowing the learners to design and implement efficient, optimal, and trouble-free multiservice networks.

Learner Skills and Knowledge

This subtopic lists the skills and knowledge that learners must possess to benefit fully from the course. The subtopic also includes recommended Cisco learning offerings that learners should first complete to benefit fully from this course.

Learner Skills and Knowledge

- Interconnecting Cisco Network Devices (ICND)

OR

- Cisco Certified Network Associate (CCNA)

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-3

Course Goal and Objectives

This topic describes the course goal and objectives.

Course Goal

“To identify, describe, and correctly implement the appropriate QoS mechanisms required to create an effective administrative policy providing QoS”

Implementing Cisco Quality of Service (QoS) v2.3

© 2009 Cisco Systems, Inc. All rights reserved. QoS v2.3-4

Upon completing this course, you will be able to meet these objectives:

- Explain the need to implement QoS and methods for implementing and managing QoS
- Identify and describe different models used for ensuring QoS in a network and explain key QoS mechanisms used to implement the models
- Explain the use of MQC and AutoQoS to implement QoS on the network
- Classify and mark network traffic to implement a policy according to QoS requirements
- Use Cisco QoS queuing mechanisms to manage network congestion
- Use Cisco QoS congestion avoidance mechanisms to reduce the effects of congestion on the network
- Use Cisco QoS traffic policing and traffic-shaping mechanisms to effectively limit the rate of network traffic
- Use Cisco link efficiency mechanisms to improve the bandwidth efficiency of low-speed WAN links
- Select the most appropriate QoS mechanisms for providing QoS using Cisco best practices in service provider and enterprise networks

Course Flow

This topic presents the suggested flow of the course materials.

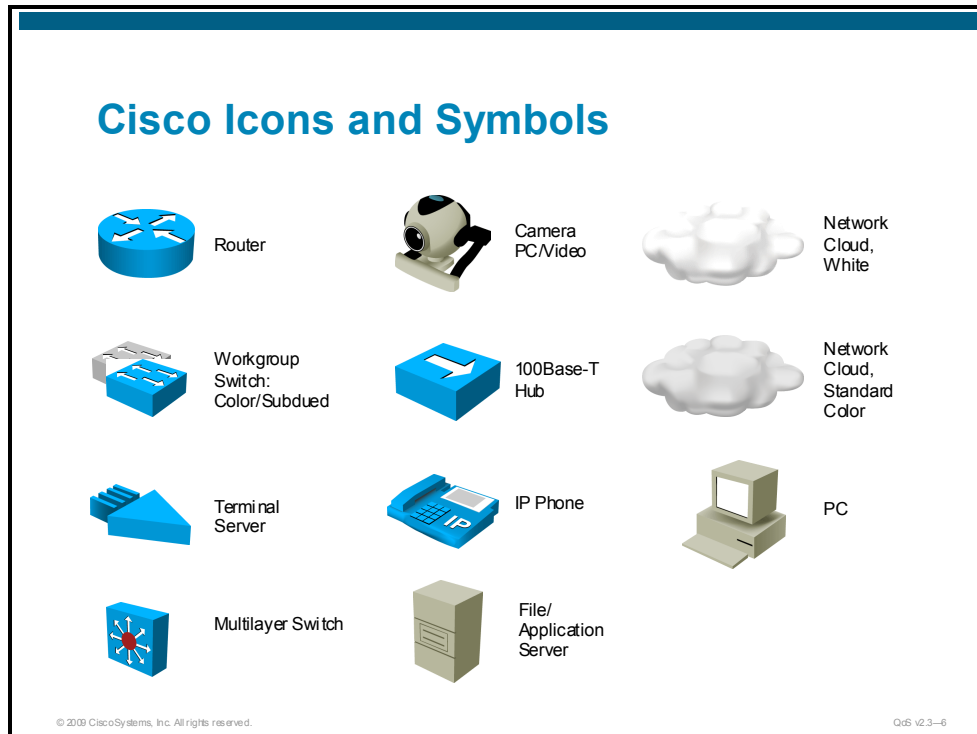
Course Flow					
Day 1		Day 2		Day 3	
Day 4		Day 5			
A M	Course Introduction	Introduction to Modular QoS CLI and AutoQoS (Cont.)	Congestion Management	Congestion Avoidance	Link Efficiency Mechanisms
	Introduction to QoS	Classification and Marking			
Lunch					
P M	Building Blocks of QoS (Cont.)				
	Introduction to Modular QoS CLI and AutoQoS	Classification and Marking (Cont.)	Congestion Management (Cont.)	Traffic Policing and Shaping	QoS Best Practices

© 2009 Cisco Systems, Inc. All rights reserved. QoS v2.3-6

The schedule reflects the recommended structure for this course. This structure allows enough time for the instructor to present the course information and for you to work through the lab activities. The exact timing of the subject materials and labs depends on the pace of your specific class.

Additional References

This topic presents the Cisco icons and symbols used in this course, as well as information on where to find additional technical references.



Cisco Glossary of Terms

For additional information on Cisco terminology, refer to the *Cisco Internetworking Terms and Acronyms* glossary of terms at <http://www.cisco.com/univercd/cc/td/doc/cisintwk/ita/index.htm>.

Your Training Curriculum

This topic presents the training curriculum for this course.

Cisco Career Certifications



www.cisco.com/go/certifications

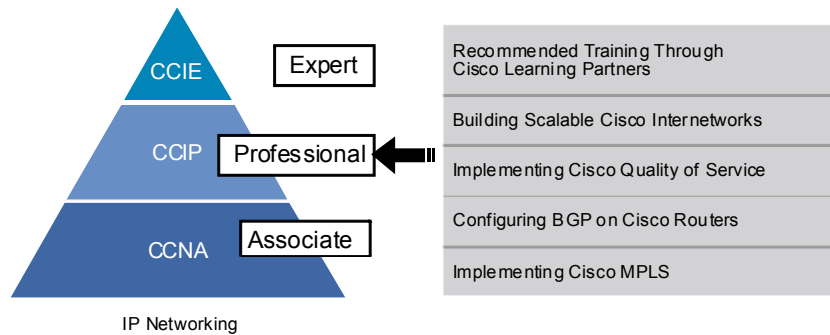
© 2009 Cisco Systems, Inc. All rights reserved. QoS v2.3-7

You are encouraged to join the Cisco Certification Community, a discussion forum open to anyone holding a valid Cisco Career Certification (such as Cisco CCIE[®], CCNA[®], CCDA[®], CCNP[®], CCDP[®], CCIP[®], CCVP[™], or CCSP[™]). It provides a gathering place for Cisco certified professionals to share questions, suggestions, and information about Cisco Career Certification programs and other certification-related topics. For more information, visit www.cisco.com/go/certifications.

Cisco Career Certifications: Cisco Certified Internetwork Professional (CCIP)

Expand Your Professional Options and Advance Your Career

Professional-level recognition in IP networking



www.cisco.com/go/certifications

© 2009 Cisco Systems, Inc. All rights reserved.

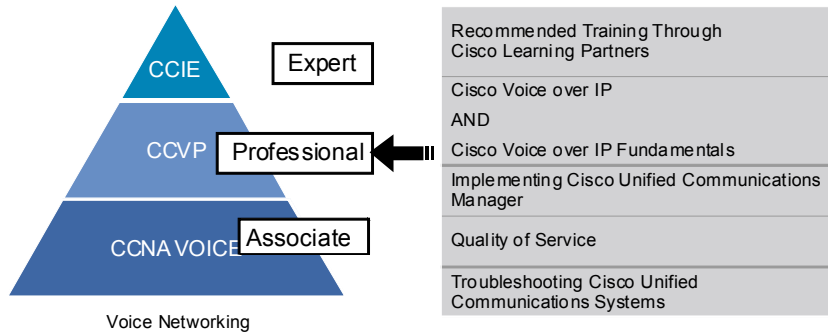
QoS v2.3-8

Cisco Career Certifications: Cisco Certified Voice Professional (CCVP)

Cisco Unified Communications Manager Path

Expand Your Professional Options and Advance Your Career

Professional-level recognition in voice networking



© 2009 Cisco Systems, Inc. All rights reserved.

www.cisco.com/go/certifications

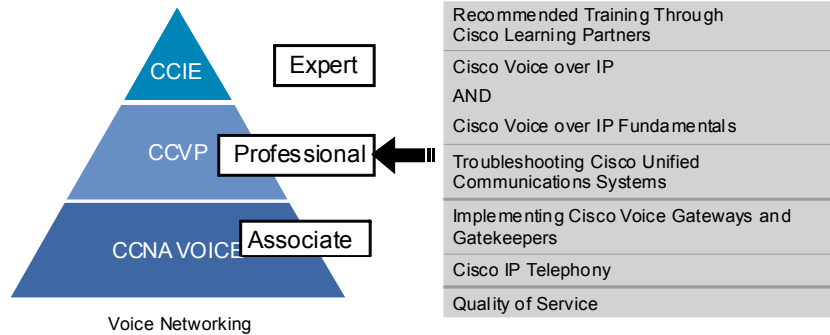
QoS v2.3-0

Cisco Career Certifications: Cisco Certified Voice Professional (CCVP)

Cisco Unified Call Manager Path

Expand Your Professional Options and Advance Your Career

Professional-level recognition in voice networking



© 2009 Cisco Systems, Inc. All rights reserved.

www.cisco.com/go/certifications

QoS v23-10

Your Training Curriculum

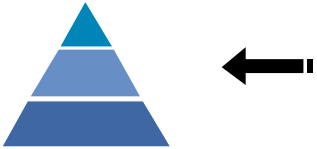
This topic presents the training curriculum for this course.

Cisco Specialist Certification Path

Enhance Your Cisco Certifications—and Validate Your Areas of Expertise

Cisco Quality of Service Specialists

Cisco IP Telephony Design Specialist



Recommended Training Through Cisco Learning Partners
Prerequisite: Valid CCDA Certification
Cisco Lifecycle Services for Advanced Unified Communications
Cisco Unified Communications Architecture and Design
Implementing Cisco Quality of Service

www.cisco.com/go/certifications

© 2010 Cisco Systems, Inc. All rights reserved. QoSv23—11

Cisco Specialist certifications demonstrate significant competency in specific technology areas, solutions, or job roles. Individuals who have earned an associate-level career certification or higher are eligible to become qualified in these focused areas. With one or more specialist certifications, network professionals can better align their core expertise with current industry needs. For more information on Cisco Specialist certifications, visit www.cisco.com/go/certifications.

Introduction to QoS

Overview

As user applications continue to drive network growth and evolution, demand to support different types of traffic is also increasing. Different types of applications with differing network requirements create a need for administrative policies mandating how individual applications are to be treated by the network. Network traffic from business-critical applications must be protected from other types of traffic. Requests from business-critical and delay-sensitive applications must be serviced with priority. The employment and enforcement of quality of service (QoS) policies within a network play an essential role in enabling network administrators and architects to meet networked application demands. QoS is a crucial element of any administrative policy that mandates how to handle application traffic on a network. This module introduces the concept of QoS, explains key issues of networked applications, and describes different methods for implementing QoS.

Module Objectives

Upon completing this module, you will be able to explain the need to implement QoS and explain methods for implementing and managing QoS. This ability includes being able to meet these objectives:

- Identify problems that could lead to poor quality of service and provide solutions to those problems
- Explain the term QoS and the key steps to implementing QoS on a converged network
- List methods for implementing QoS

Understanding the Need for QoS

Overview

Networks must provide secure, predictable, measurable, and, sometimes, guaranteed services. Network administrators and architects can better achieve this performance from the network by managing delay, delay variation (jitter), bandwidth provisioning, and packet loss parameters with quality of service (QoS) techniques. This lesson introduces the concept of a converged network, identifies four problems that could lead to poor quality of service, and describes solutions to those problems.

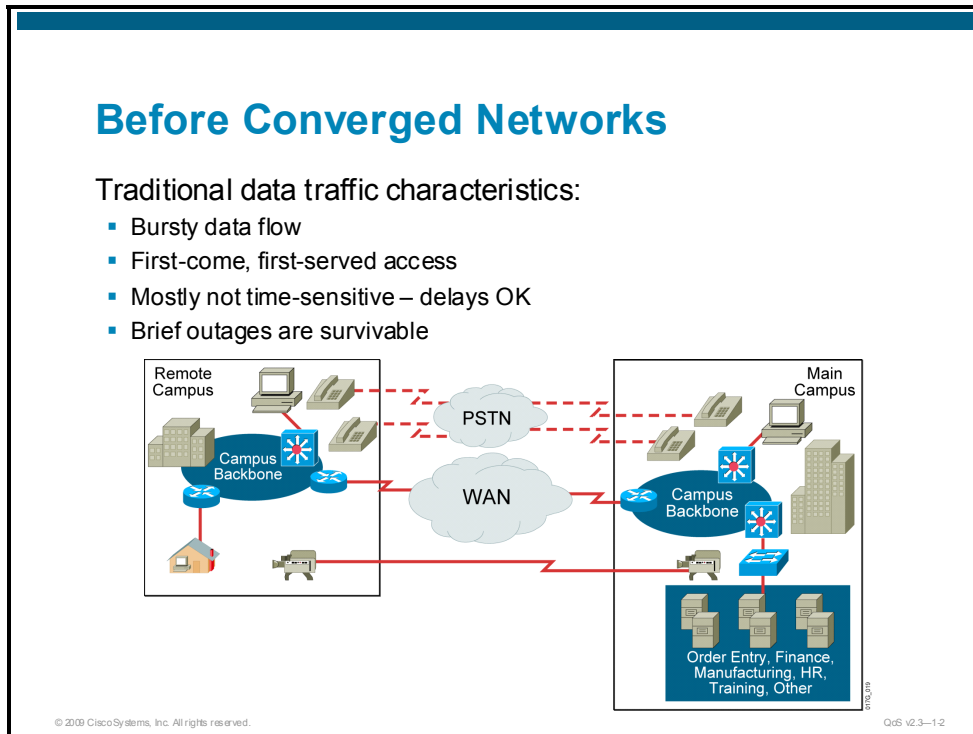
Objectives

Upon completing this lesson, you will be able to identify problems that could lead to poor QoS and the solutions to those problems. This ability includes being able to meet these objectives:

- Describe a converged IP network supporting voice, video, and data traffic, and explain why QoS was not important before networks converged
- Describe the four key quality issues with converged networks: lack of bandwidth, end-to-end delay (fixed and variable), variation of delay (jitter), and packet loss
- Explain how a lack of bandwidth can adversely impact QoS in a network and ways to effectively increase bandwidth on a link
- Explain how end-to-end delay can adversely impact QoS in a network and ways to effectively reduce delay
- Explain how packet loss can adversely impact QoS in a network and ways to manage packet loss so that QoS is not affected

Converged Networks

This topic describes a converged IP network supporting voice, video, and data traffic and explains why QoS was not important before networks converged.



Before networks converged, network engineering focused on connectivity. The rates at which data came onto the network resulted in bursty data flows. Data, arriving in packets, tried to grab as much bandwidth as it could at any given time. Access was on a first-come, first-served basis. The data rate available to any one user varied depending on the number of users accessing the network at any given time.

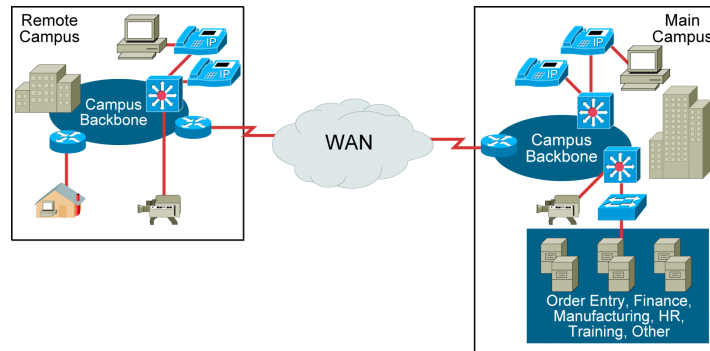
The protocols that have been developed have adapted to the bursty nature of data networks, and brief outages are survivable. For example, when you retrieve email, a delay of a few seconds is generally not noticeable. A delay of minutes is annoying, but not serious.

Traditional networks also had requirements for applications such as data, video, and systems network architecture (SNA). Since each application had different traffic characteristics and requirements, network designers deployed nonintegrated networks. These nonintegrated networks were designed to carry a specific type of traffic: data network, SNA network, voice network, and video network.

After Converged Networks

Converged traffic characteristics:

- Constant small-packet voice flow competes with bursty data flow
- Critical traffic must get priority
- Voice and video are time-sensitive
- Brief outages not acceptable



The figure illustrates a converged network in which voice, video, and data traffic use the same network facilities. Merging these different traffic streams with dramatically differing requirements can lead to a number of problems.

Although packets carrying voice traffic are typically very small, they cannot tolerate delay and delay variation as they traverse the network. Voices will break up and words will become incomprehensible.

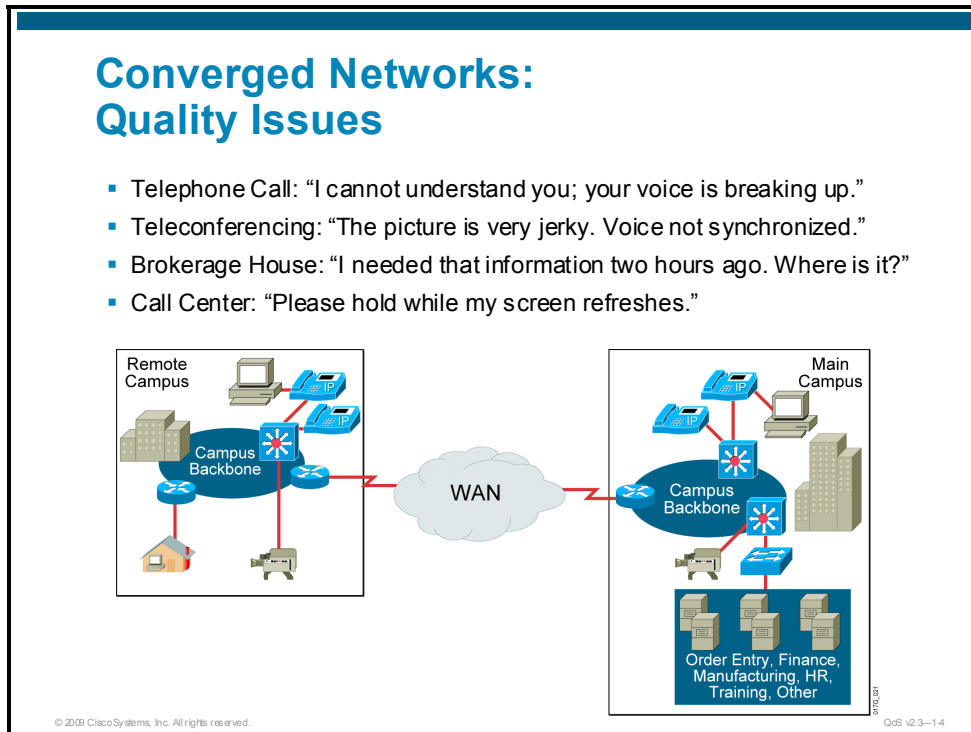
On the other hand, packets carrying file transfer data are typically large and can survive delays and drops. It is possible to retransmit part of a dropped data file, but it is not feasible to retransmit a part of a voice conversation.

The constant, small-packet voice flow competes with bursty data flows. Unless some mechanism mediates the overall flow, voice quality will be severely compromised at times of network congestion. The critical voice traffic must get priority. Voice and video traffic is very time-sensitive. It cannot be delayed and it cannot be dropped, or the resulting quality of voice and video will suffer.

Finally, converged networks cannot fail. While a file transfer or email packet can wait until the network recovers, voice and video packets cannot wait. Even a brief network outage on a converged network can seriously disrupt business operations.

Converged Networks Quality Issues

This topic describes the four key quality issues with converged networks: lack of bandwidth, end-to-end delay (fixed and variable), variation of delay (jitter), and packet loss.



With inadequate preparation of the network, voice transmission is choppy or unintelligible. Gaps in speech are particularly troublesome where pieces of speech are interspersed with silence. In voice-mail systems, this silence is a problem. For example, when 68614 is dialed and the gaps in speech are actually gaps in the tone, 68614 becomes 6688661144, because the gaps in speech are perceived as pauses in the touch tones.

Poor caller interactivity is the consequence of delay and causes these problems:

- **Echo:** Echo is caused by the signal reflecting the speaker voice from the far-end telephone equipment back into the speaker ear.
- **Talker overlap:** Talker overlap is caused when one-way delay becomes greater than 250 ms. When this delay occurs, the speech of one talker overlaps the speech of the other talker, resulting in a "walkie-talkie" call mode.

Disconnected calls are the worst cases. If there are long gaps in speech, the parties will hang up. If there are signaling problems, calls are disconnected. Such events are completely unacceptable in voice communications, yet are quite common for an inadequately prepared data network that is attempting to carry voice.

Multimedia streams, such as those used in IP telephony or videoconferencing, may be extremely sensitive to delivery delays and may create unique QoS demands on the underlying networks that carry them. When packets are delivered using the Best-Effort delivery model, they may not arrive in order, in a timely manner, or at all. The result is unclear pictures, jerky and slow movement, and sound that is out of synchronization with the image.

Converged Networks: Quality Issues (Cont.)



- **Lack of bandwidth:** Multiple flows compete for a limited amount of bandwidth.
- **End-to-end delay (fixed and variable):** Packets have to traverse many network devices and links that add up to the overall delay.
- **Variation of delay (jitter):** Sometimes there is a lot of other traffic, which results in more delay.
- **Packet loss:** Packets may have to be dropped when a link is congested.

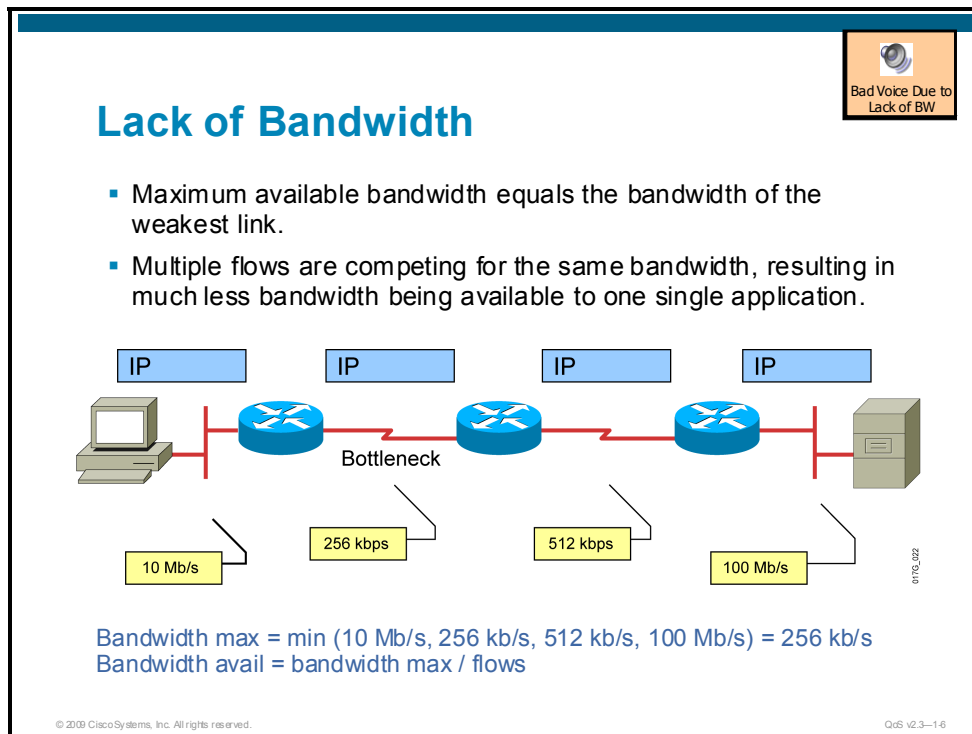
© 2009 Cisco Systems, Inc. All rights reserved. QoS v2.3-1-5

The four big problems facing converged enterprise networks are as follows:

- **Bandwidth capacity:** Large graphics files, multimedia uses, and increasing use of voice and video cause bandwidth capacity problems over data networks.
- **End-to-end delay (both fixed and variable):** Delay is the time it takes for a packet to reach the receiving endpoint after being transmitted from the sending endpoint. This period of time is called the “end-to-end delay,” and consists of two components:
 - **Fixed network delay:** Two types of fixed delays are serialization and propagation delays. Serialization is the process of placing bits on the circuit. The higher the circuit speed, the less time it takes to place the bits on the circuit. Therefore, the higher the speed of the link, the less serialization delay is incurred. Propagation delay is the time it takes for frames to transit the physical media.
 - **Variable network delay:** Processing delay is a type of variable delay, and is the time required by a networking device to look up the route, change the header, and complete other switching tasks. In some cases, the packet also must be manipulated, as, for example, when the encapsulation type or the hop count must be changed. Each of these steps can contribute to the processing delay.
- **Variation of delay (also called jitter):** Jitter is the delta, or difference, in the total end-to-end delay values of two voice packets in the voice flow.
- **Packet loss:** Loss of packets is usually caused by congestion in the WAN, resulting in speech dropouts or a stutter effect if the playout side tries to accommodate by repeating previous packets.

Available Bandwidth

This topic describes how a lack of bandwidth can adversely impact QoS in a network and describes ways to effectively increase bandwidth on a link.

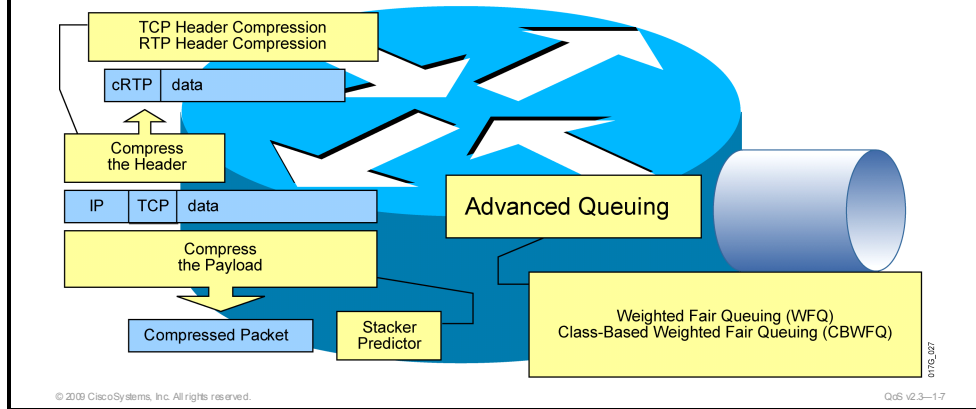


The figure illustrates an empty network with four hops between a server and a client. Each hop is using different media with a different bandwidth. The maximum available bandwidth is equal to the bandwidth of the weakest (slowest) link.

The calculation of the available bandwidth, however, is much more complex in cases where multiple flows traverse the network. The calculation of the available bandwidth in the illustration is an approximation.

Ways to Increase or Manage Available Bandwidth

- Upgrade the link: the best solution but also the most expensive.
- Forward the important packets first.
- Compress the payload of Layer 2 frames (it takes time).
- Compress IP packet headers.



The best way to increase bandwidth is to increase the link capacity to accommodate all applications and users, with some extra bandwidth to spare. Although this solution sounds simple, increasing bandwidth is expensive and takes time to implement. There are often technological limitations in upgrading to a higher bandwidth.

Another option is to classify traffic into QoS classes and prioritize traffic according to importance. Voice and business-critical traffic should get sufficient bandwidth to support their application requirements, voice should get prioritized forwarding, and the least important traffic should get whatever unallocated bandwidth is remaining. A variety of mechanisms such as these are available in Cisco IOS QoS Software to provide bandwidth guarantees:

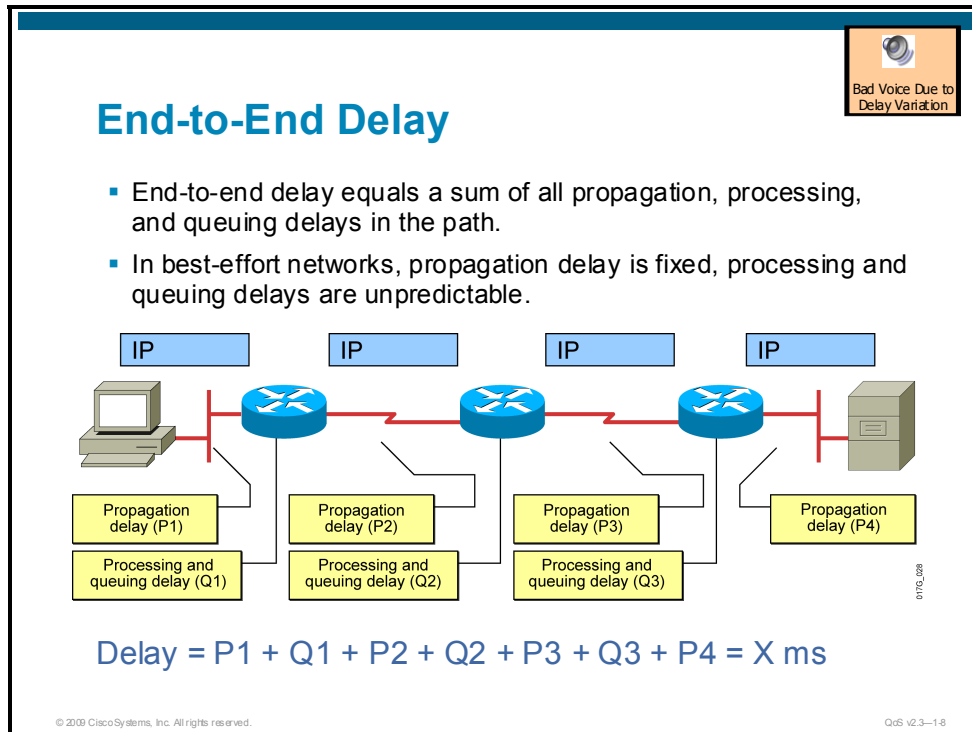
- Priority queuing (PQ) or custom queuing (CQ)
- Modified deficit round robin (MDRR) (on Cisco 12000 Series routers)
- Distributed type of service (ToS)-based and QoS group-based weighted fair queuing (WFQ) (on Cisco 7x00 Series routers)
- Class-based weighted fair queuing (CBWFQ)
- Low-latency queuing (LLQ)

Optimizing link usage by compressing the payload of frames (virtually) increases the link bandwidth. Compression, on the other hand, also increases delay because of the complexity of compression algorithms. Using hardware compression can accelerate packet payload compressions. Stacker and Predictor are two compression algorithms that are available in Cisco IOS Software.

Another link efficiency mechanism is header compression. Header compression is especially effective in networks in which most packets carry small amounts of data (that is, where payload-to-header ratio is small). Typical examples of header compression are TCP header compression and Real-Time Transport Protocol (RTP) header compression.

End-to-End Delay

This topic describes how end-to-end delay can adversely impact QoS in a network and describes ways to effectively reduce delay.



The figure illustrates the impact that a network has on the end-to-end delay of packets going from one end of the network to the other. Each hop in the network adds to the overall delay because of these factors:

- Propagation delay is caused by the speed of light traveling in the media; for example, the speed of light traveling in fiber optics or copper media.
- Serialization delay is the time it takes to clock all the bits in a packet onto the wire. This is a fixed value that is a function of the link bandwidth.
- There are processing and queuing delays within a router, which can be caused by a wide variety of conditions.

Propagation delay is generally ignored but it can be significant; for example, about 40 ms coast-to-coast, over optical. Internet Control Message Protocol (ICMP) echo (ping) is one way to measure the round-trip time of IP packets in a network.

Example: Effects of Delay

A customer has a router in New York and a router in San Francisco, each connected by a 128-kbps WAN link. The customer sends a 66-byte voice frame across the link. To transmit the frame (528 bits), it will take 4.125 ms to clock out (serialization delay). However, the last bit will not arrive until 40 ms *after* it clocks out (propagation delay). The total delay equals 44.125 ms.

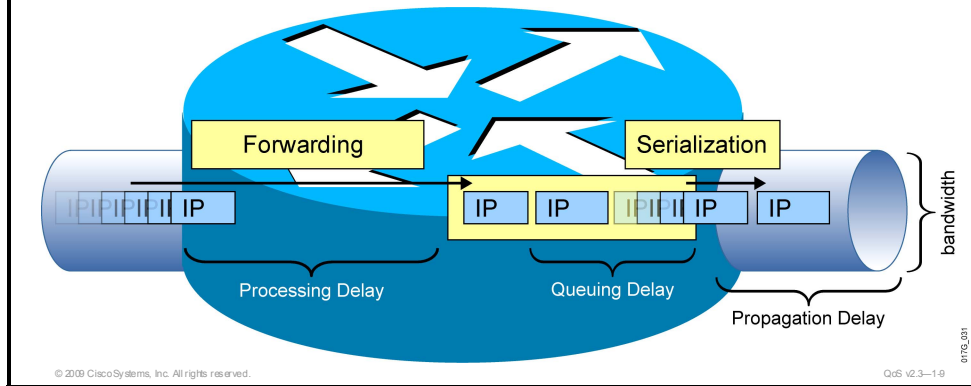
Now, change the circuit to a T1. To transmit the frame (528 bits), it will take 0.344 ms to clock out (serialization delay). However, the last bit will not arrive until 40 ms after transmission

(propagation delay) for a total delay of 40.344 ms. In this case, the significant factor is propagation delay. In the same situation—but for a link between Seattle and San Francisco—serialization delay remains the same and propagation delay drops to about 6 ms, making 528 bits take 10.125 ms (on a 128-kbps link), and a total delay of 6.344 (on a T1 link).

Both serialization and propagation delays must be taken into account in determining total delay.

Types of Delay

- **Processing Delay:** The time it takes for a router to take the packet from an input interface, examine it, and put it into the output queue of the output interface
- **Queuing Delay:** The time a packet resides in the output queue of a router
- **Serialization Delay:** The time it takes to place the “bits on the wire”
- **Propagation Delay:** The time it takes to transmit a packet

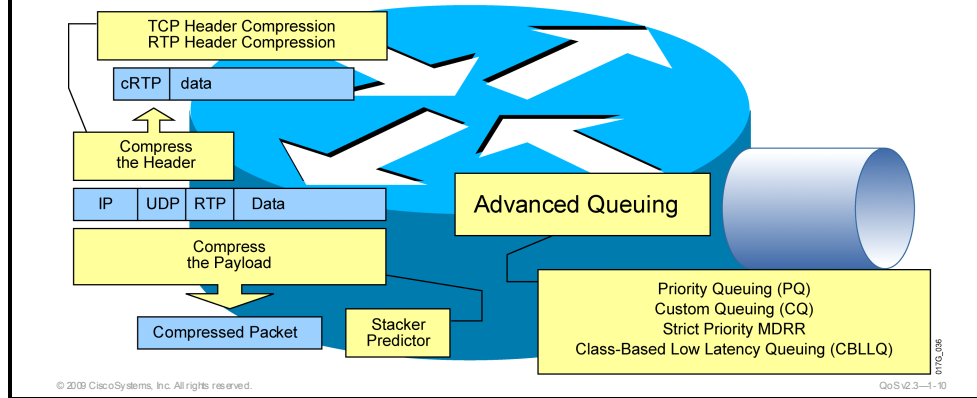


In summary, there are four types of delay, as follows:

- **Processing delay:** The time it takes for a router to take the packet from an input interface and put the packet into the output queue of the output interface. The processing delay depends on these factors:
 - CPU speed
 - CPU utilization
 - IP switching mode
 - Router architecture
 - Configured features on both input and output interfaces
- **Queuing delay:** The time a packet resides in the output queue of a router. Queuing delay depends on the number of and sizes of packets already in the queue, the bandwidth of the interface, and the queuing mechanism.
- **Serialization delay:** The time it takes to place a frame on the physical medium for transport.
- **Propagation delay:** The time it takes to transmit a packet, which usually depends on the type of media interface.

Ways to Reduce Delay

- Upgrade the link; the best solution but also the most expensive.
- Forward the important packets first.
- Compress the payload of Layer 2 frames (it takes time).
- Compress IP packet headers.



Assuming that a router is powerful enough to make a forwarding decision rapidly, most processing, queuing, and serialization delay is influenced by these factors:

- Average length of the queue
- Average length of packets in the queue
- Link bandwidth

These approaches allow you to accelerate packet dispatching of delay-sensitive flows:

- **Increasing link capacity:** Sufficient bandwidth causes queues to shrink so that packets do not wait long before transmittal. More bandwidth reduces serialization time. This can be an unrealistic approach because of the costs associated with the upgrade.
- **Prioritizing delay-sensitive packets:** This is a more cost-effective approach. PQ, CQ, strict-priority or alternate-priority queuing within the MDRR (on Cisco 12000 Series routers), and LLQ each have preemptive queuing capabilities.
- **Compressing the payload:** Payload compression reduces the size of packets, thereby virtually increasing link bandwidth. Compressed packets are smaller and take less time to transmit. Compression uses complex algorithms that take time and add to delay. This approach is not used to provide low-delay propagation of packets.
- **Compressing the packet header:** Header compression is not as CPU-intensive as payload compression and you can use it with other mechanisms to reduce delay. Header compression is especially useful for voice packets that have a bad payload-to-header ratio, which you can improve by reducing the header of the packet (RTP header compression). By minimizing delay, you can also reduce jitter (delay is more predictable).

Packet Loss

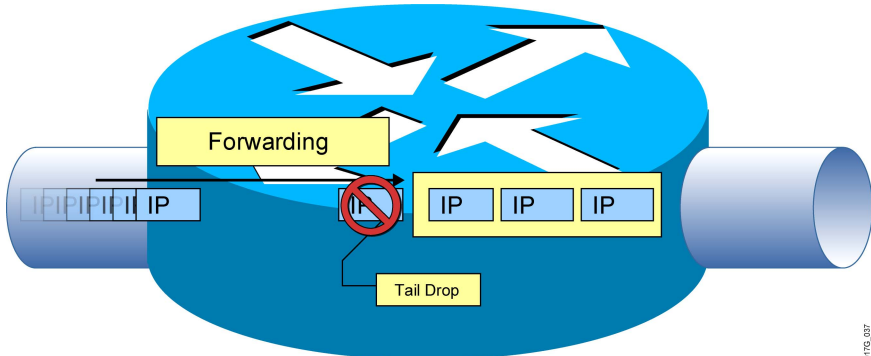
This topic describes how packet loss can adversely impact QoS in a network and describes ways to manage packet loss so that QoS is not affected.



Bad Voice Due
to Packet Loss

Packet Loss

- **Tail drops** occur when the output queue is full. These are common drops, which happen when a link is congested.
- Many other types of drops exist, usually the result of router congestion, that are uncommon and may require a hardware upgrade (input drop, ignore, overrun, frame errors).



The diagram shows a blue router with a large white 'X' on its front. On the left, a blue pipe represents an input link with four packets (labeled 'IP') in a queue. A yellow box labeled 'Forwarding' has an arrow pointing to a yellow box on the right, which represents the output queue and contains three packets (labeled 'IP'). A red circle with a diagonal slash is positioned between the forwarding box and the output queue. A yellow box labeled 'Tail Drop' has an arrow pointing to this red circle, indicating that packets are being dropped at the tail of the output queue.

© 2008 Cisco Systems, Inc. All rights reserved.

0176_037
QoS v2.3-1-11

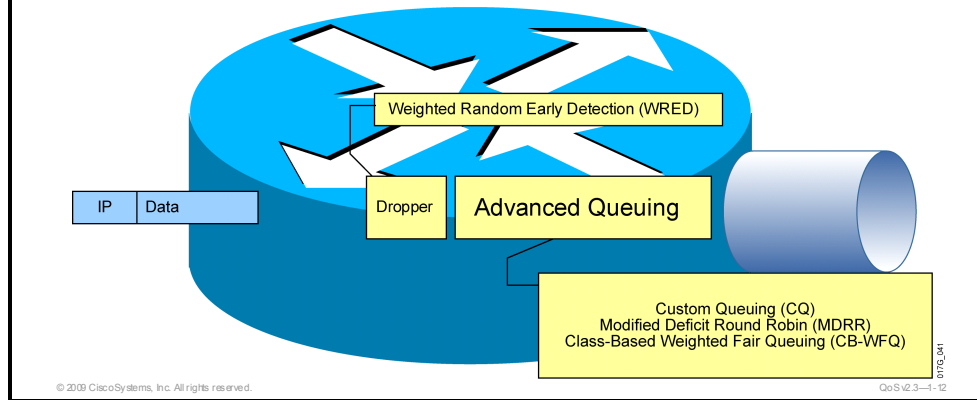
The usual packet loss occurs when routers run out of buffer space for a particular interface output queue. The figure illustrates a full interface output queue, which causes newly arriving packets to be dropped. The term used for such drops is simply “output drop” or “tail drop” (packets are dropped at the tail of the queue).

Routers might also drop packets for these other less common reasons:

- **Input queue drop:** The main CPU is congested and cannot process packets (the input queue is full).
- **Ignore:** The router ran out of buffer space.
- **Overrun:** The CPU is congested and cannot assign a free buffer to the new packet.
- **Frame errors:** There is a hardware-detected error in a frame—cyclic redundancy check (CRC), runt, giant.

Ways to Prevent Packet Loss

- Upgrade the link; the best solution but also the most expensive.
- Guarantee enough bandwidth to sensitive packets.
- Prevent congestion by randomly dropping less important packets before congestion occurs.



Packet loss is usually the result of congestion on an interface. Most applications that use TCP experience slowdown because TCP adjusts to the network resources. Dropped TCP segments cause TCP sessions to reduce their window sizes. There are some other applications that do not use TCP and cannot handle drops.

You can follow these approaches to prevent drops of sensitive applications:

- Increase link capacity to ease or prevent congestion.
- Guarantee enough bandwidth and increase buffer space to accommodate bursts of fragile applications. There are several mechanisms available in Cisco IOS QoS Software that can guarantee bandwidth and provide prioritized forwarding to drop-sensitive applications, such as these mechanisms:
 - Priority queuing (PQ)
 - Modified deficit round robin (MDRR) (on Cisco 12000 Series routers)
 - IP RTP priority
 - Class-based weighted fair queuing (CBWFQ)
 - Low latency queuing (LLQ)
- Prevent congestion by dropping other packets before congestion occurs. You can use weighted random early detection (WRED) to start dropping other packets before congestion occurs.

These are some other mechanisms that you can use to prevent congestion:

- **Traffic shaping:** Traffic shaping delays packets instead of dropping them and includes generic traffic shaping, Frame Relay traffic shaping (FRTS), and class-based shaping.
- **Traffic policing:** Traffic policing, including committed access rate (CAR) and class-based policing, can limit the rate of less-important packets to provide better service to drop-sensitive packets.

Summary

This topic summarizes the key points discussed in this lesson.

Summary

- Traditional networks use a separate network for delay- and jitter-sensitive voice traffic. The voice network is engineered to support the required number of calls. For data traffic, protocols such as TCP have been developed to adapt to the bursty nature of a data network.
- Converged networks that support voice, video, and data create new requirements for managing network traffic. QoS meets those requirements.
- Converged networks suffer from different quality issues, including lack of adequate bandwidth, end-to-end and variable delay, and lost packets.
- Lack of bandwidth causes networks to experience different types of delay, including processing delay, queuing delay, serialization delay, and propagation delay.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3—1-13

Summary (Cont.)

- End-to-end delay is the sum of all propagation, processing, and queuing delays.
- Packet loss can cause applications that use TCP to slow down. Applications that do not use TCP, such as voice traffic, will experience poor voice quality if the packet loss is too excessive.
- Ways to increase the available bandwidth, and decrease the delay and packet loss, include:
 - Increasing the link capacity
 - Using QoS mechanisms such as queuing, WRED, traffic policing and shaping, and link efficiency mechanisms such as payload and header compression

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3—1-14

Understanding QoS

Overview

As networks increasingly converge to support voice, video, and data traffic, there is a growing need for quality of service (QoS). Network administrators and architects must be able to develop QoS systems to accommodate this new traffic. This lesson describes the basic concepts and key terminology of QoS and the three key steps involved in implementing a QoS policy.

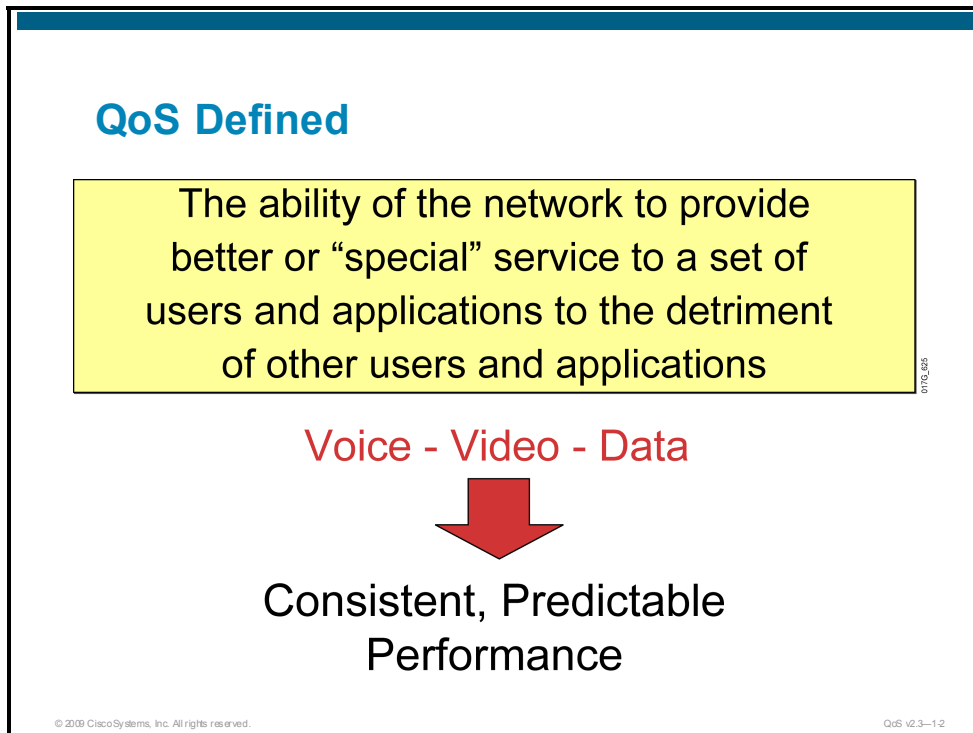
Objectives

Upon completing this lesson, you will be able to define QoS and describe the key steps for implementing QoS on a converged network. This ability includes being able to meet these objectives:

- Define QoS with respect to traffic in a network
- Explain the three key steps involved in implementing a QoS policy on a network
- Describe how traffic is identified on a network and describe basic QoS requirements
- Explain how to divide traffic into traffic classes
- Describe how to define QoS policies after traffic classes have been defined

QoS Defined

This topic describes QoS with respect to traffic in a network.



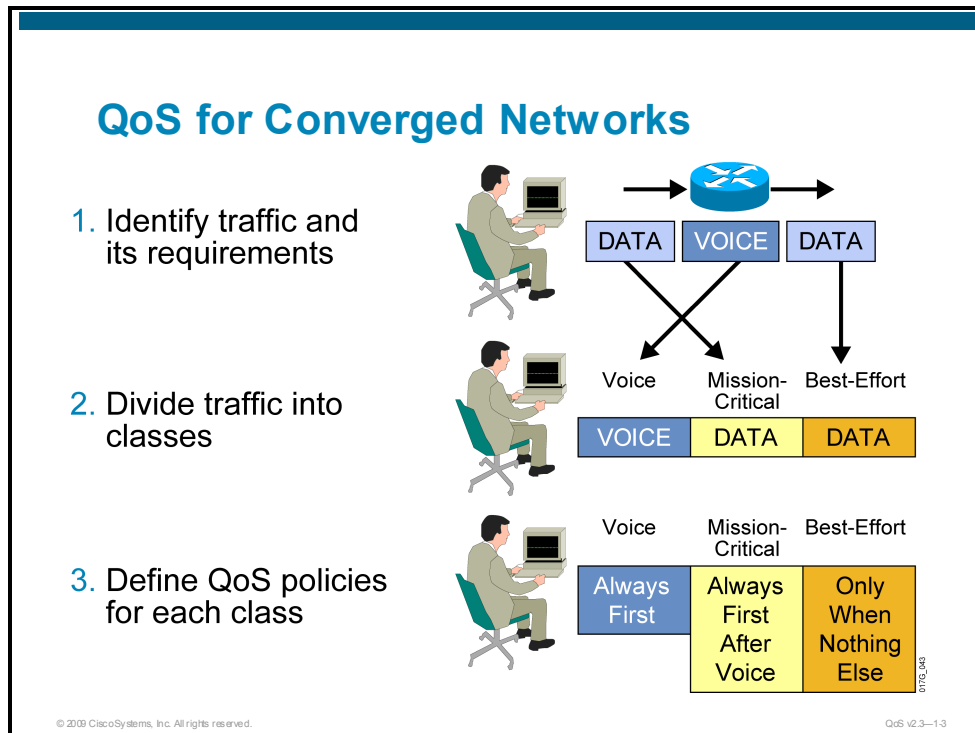
QoS is the ability of the network to provide better or “special” service to selected users and applications, to the detriment of other users and applications.

Cisco IOS QoS features enable you to control and predictably service a variety of networked applications and traffic types, thus allowing you to take advantage of a new generation of media-rich and mission-critical applications.

The goal of QoS is to provide better and more predictable network service by providing dedicated bandwidth, controlled jitter and latency, and improved loss characteristics. QoS achieves these goals by providing tools for managing network congestion, shaping network traffic, using expensive wide-area links more efficiently, and setting traffic policies across the network. QoS offers intelligent network services that, when correctly applied, help to provide consistent, predictable performance.

QoS for Converged Networks

This topic describes the three key steps involved in implementing a QoS policy on a network.



There are three basic steps involved in implementing QoS on a network:

- Step 1** Identify traffic and its requirements. Study the network to determine the type of traffic running on the network and then determine the QoS requirements for the different types of traffic.
- Step 2** Group the traffic into classes with similar QoS requirements. For example, four classes of traffic can be defined: *voice*, *high priority*, *low priority*, and *browser*.
- Step 3** Define QoS policies that will meet the QoS requirements for each traffic class.

Example: Three Steps to Implementing QoS on a Network

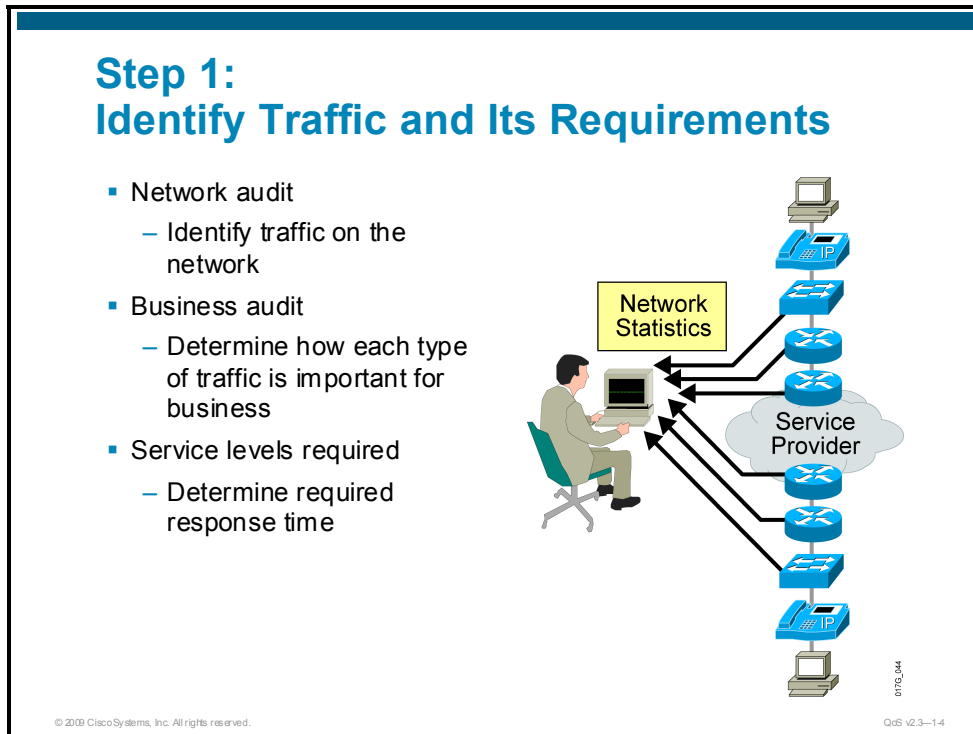
In a typical network, voice will always require absolute minimal delay. Some data associated with key applications will require very low delay (transaction-based data used in airline reservations or online banking applications). Other types of data can tolerate a great deal of delay (file transfers and email). Nonbusiness network surfing can also be delayed or even prohibited.

A one-to-one mapping between traffic classes and QoS policies need not be made. For example, three QoS policies could be implemented to meet the requirements of the four traffic classes defined above:

- **NoDelay:** Assign to voice traffic
- **BestService:** Assign to high-priority traffic
- **Whenever:** Assign to both the low-priority and browser traffic

QoS Requirements

This topic describes how traffic is identified on a network and describes elemental QoS requirements.



The first step in implementing QoS is identifying the traffic on the network and determining QoS requirements for the traffic.

The next step is determining the QoS problems of users. Measure the traffic on the network during congested periods. Conduct CPU utilization assessment on each of their network devices during busy periods to determine where problems might be occurring.

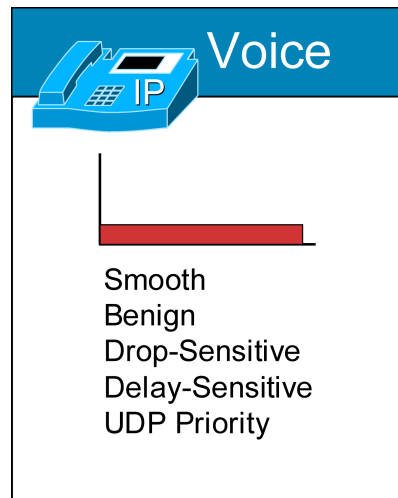
Next, determine the business model and goals, and obtain a list of business requirements, in order to define the number of classes so that you can determine the business requirements for each traffic class.

Finally, define the service levels required by different traffic classes in terms of response time and availability. What is the impact on business if a transaction is delayed by two or three seconds? Can file transfers wait until the network is quiescent?

QoS Traffic Requirements: Voice

- Latency = 150 ms*
- Jitter = 30 ms*
- Loss = 1%*
- 17–106 kb/s guaranteed priority bandwidth per call
- 150 b/s (+ Layer 2 overhead) guaranteed bandwidth for voice-control traffic per call

*One-way requirements



Voice traffic has extremely stringent QoS requirements. Voice traffic usually generates a smooth demand on bandwidth and has minimal impact on other traffic as long as the voice traffic is managed.

While voice packets are typically small (60 to 120 bytes), they cannot tolerate delay or drops. The result of delays and drops are poor, and often unacceptable, voice quality. Because drops cannot be tolerated, User Datagram Protocol (UDP) is used to package voice packets, because TCP retransmit capabilities have no value.

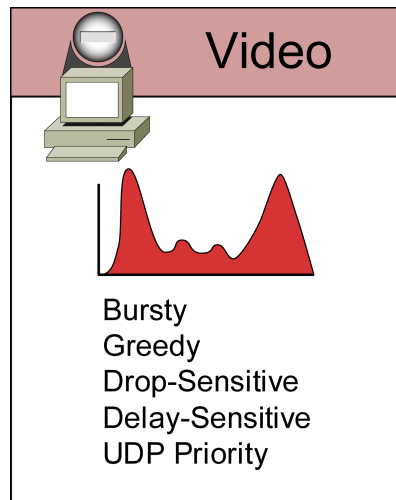
Voice packets can tolerate no more than a 150-ms delay (one-way requirement) and less than 1 percent packet loss.

A typical voice call will require 17 to 106 kb/s of guaranteed priority bandwidth plus an additional 150 b/s per call for voice-control traffic. Multiplying these bandwidth requirements times the maximum number of calls expected during the busiest time period will provide an indication of the overall bandwidth required for voice traffic.

QoS Requirements: Videoconferencing

- Latency = 150 ms*
- Jitter = 30 ms*
- Loss = 1%*
- Minimum priority bandwidth guarantee required is:
 - Video stream + 20%
 - For example, a 384-kb/s stream would require 460 kb/s of priority bandwidth

*One-way requirements



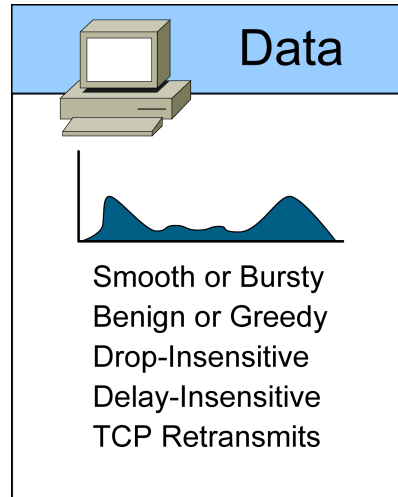
Videoconferencing applications also have stringent QoS requirements very similar to voice. But videoconferencing traffic is often bursty and greedy in nature and, as a result, can impact other traffic. Therefore, it is important to understand the videoconferencing requirements for a network and to provision carefully for it.

The minimum bandwidth for a videoconferencing stream would require the actual bandwidth of the stream (dependent upon the type of videoconferencing codec being used) plus some overhead. For example, a 384-kb/s video stream would actually require a total of 460 kb/s of priority bandwidth.

QoS Traffic Requirements: Data

- Different applications have different traffic characteristics.
- Different versions of the same application can have different traffic characteristics.
- Classify data into relative-priority model with no more than four to five classes:

- Mission-critical apps: Locally defined critical applications
- Transactional: Interactive traffic, preferred data service
- Best-effort: Internet, email, unspecified traffic
- Less-than-best-effort (Scavenger): Napster, Kazaa, peer-to-peer applications



© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-1.7

The QoS requirements for data traffic vary greatly. Different applications may make very different demands on the network (for example, a human resources application versus an automated teller machine application). Even different versions of the same application may have varying network traffic characteristics.

While data traffic can demonstrate either smooth or bursty characteristics depending upon the application, data traffic differs from voice and video in terms of delay and drop sensitivity. Almost all data applications can tolerate some delay and generally can tolerate high drop rates.

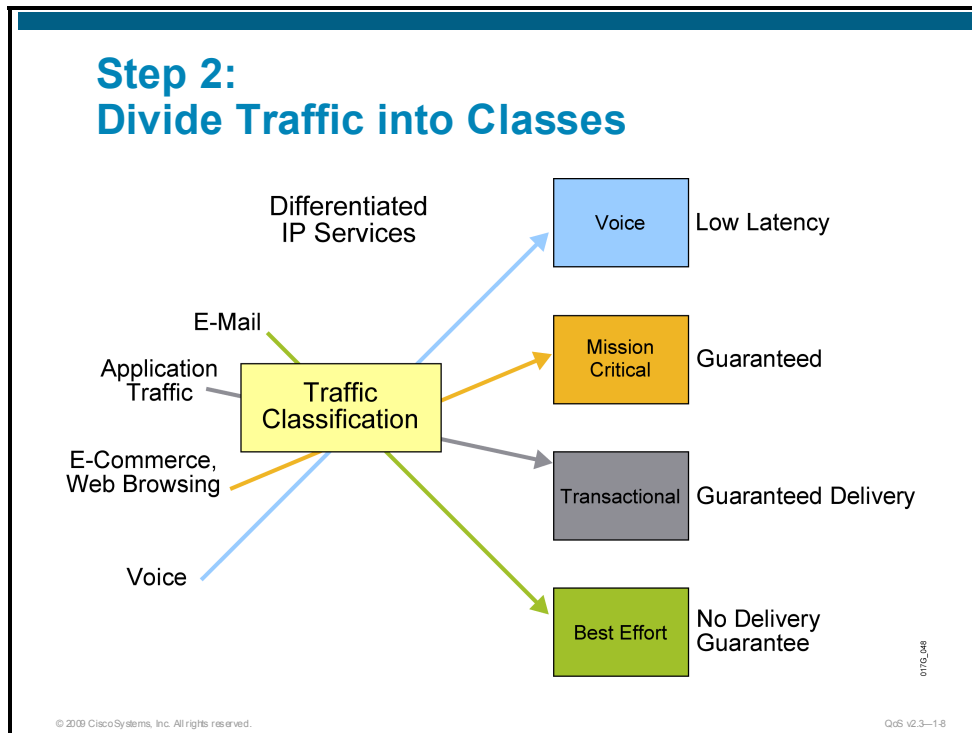
Because data traffic can tolerate drops, the retransmit capabilities of TCP become important and, as a result, many data applications use TCP.

In enterprise networks, important (business-critical) applications are usually easy to identify. Most applications can be identified based on TCP or UDP port numbers. Some applications use dynamic port numbers that, to some extent, make classifications more difficult. Cisco IOS Software supports Network-Based Application Recognition (NBAR), which you can use to recognize dynamic port applications.

It is recommended that data traffic be classified into no more than four to five classes, as described in the graphic. There will still remain additional classes for voice and video.

QoS Traffic Classes

This topic describes how to divide traffic into traffic classes.



After the majority of network traffic has been identified and measured, use the business requirements to define traffic classes.

Because of its stringent QoS requirements, voice traffic will almost always exist in a class by itself. Cisco has developed specific QoS mechanisms such as low-latency queuing (LLQ) that ensure that voice always receives priority treatment over all other traffic.

After you define the applications with the most critical requirements, you can define the remaining traffic classes using the business requirements.

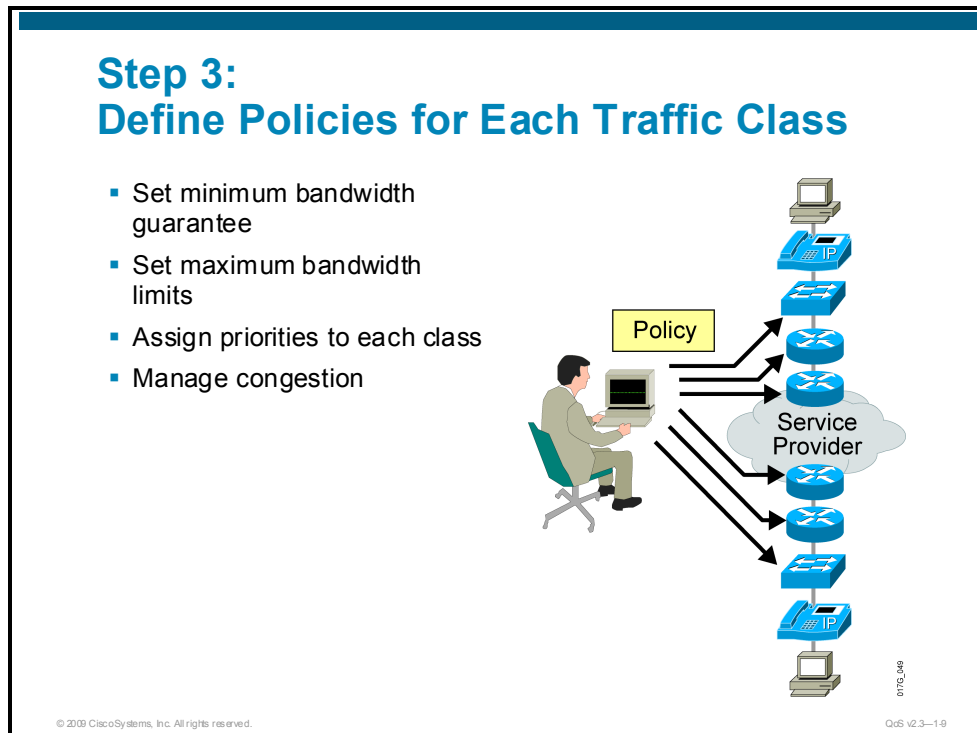
Example: Traffic Classification

A typical enterprise might define five traffic classes as follows:

- **Voice:** Absolute priority for VoIP traffic
- **Mission-critical:** Small set of locally defined critical business applications
- **Transactional:** Database access, transaction services, interactive traffic, preferred data services
- **Best-effort:** Internet, email
- **Scavenger (less-than-best-effort):** Napster, Kazaa, and other point-to-point applications

QoS Policy

This topic describes how to define QoS policies after traffic classes have been defined.



Finally, define a QoS policy for each traffic class, which involves these activities:

- Set a minimum bandwidth guarantee
- Set a maximum bandwidth limit
- Assign priorities to each class
- Use QoS technologies, such as advanced queuing, to manage congestion

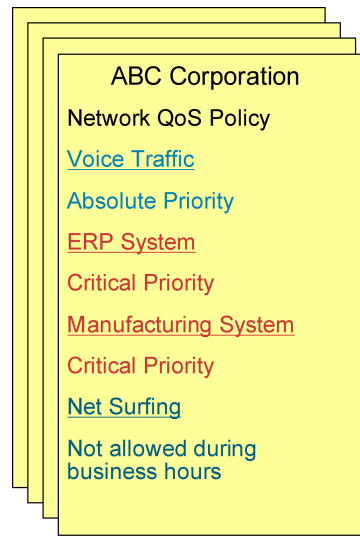
Example: Defining QoS Policies

Using the traffic classes previously defined, you can determine QoS policies as follows:

- **Voice:** Minimum bandwidth: 1 Mb/s. Use QoS marking to mark voice packets as priority level 5; use LLQ to always give voice priority.
- **Mission-critical:** Minimum bandwidth: 1 Mb/s. Use QoS marking to mark critical data packets as priority level 4; use class-based weighted fair queuing (CBWFQ) to prioritize critical class traffic flows.
- **Best-effort:** Maximum bandwidth: 500 kb/s. Use QoS marking to mark these data packets as priority level 2; use CBWFQ to prioritize best-effort traffic flows that are below mission-critical and voice.
- **Scavenger:** Maximum bandwidth: 100 kb/s. Use QoS marking to mark less-than-best-effort (scavenger) data packets as priority level 0; use weighted random early detection (WRED) to drop these packets whenever the network has a propensity for congestion.

QoS Policy

A network-wide definition of the specific levels of quality of service assigned to different classes of network traffic



© 2009 Cisco Systems, Inc. All rights reserved.

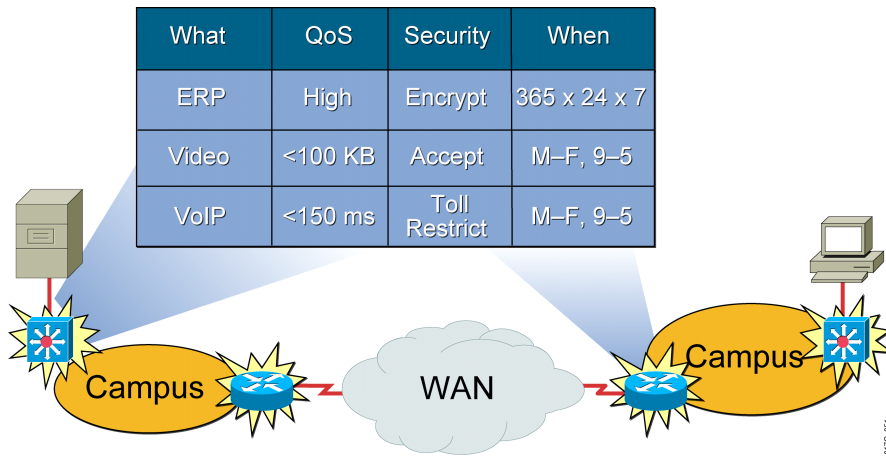
QoS v2.3—1-10

A QoS policy is a network-wide definition of the specific levels of QoS assigned to different classes of network traffic.

In a converged network, having a QoS policy is as important as having a security policy. A written and public QoS policy allows users to understand and negotiate for QoS in the network.

QoS Policy (Cont.)

Align Network Resources with Business Priorities



The figure illustrates how a QoS policy could be defined for a network.

Enterprise resource planning (ERP) applications have a high QoS priority and must be available all the time.

Video applications are guaranteed 100 KB/s of bandwidth, but can operate only between the hours of 9:00 a.m. to 5:00 p.m. (0900 to 1700) on weekdays.

Voice traffic is guaranteed less than 150-ms delay in each direction but limited to the hours of 9:00 a.m. to 5:00 p.m. (0900 to 1700) on weekdays; toll calls are completely restricted to avoid personal long-distance calls.

Summary

This topic summarizes the key points discussed in this lesson.

Summary

- QoS is the ability of the network to provide better or “special” service to users and applications.
- Building QoS requires three steps: Identify requirements, classify network traffic, and define network-wide policies for quality.
- Voice, video, and data have very different QoS requirements to run effectively on a network. These requirements affect how voice, video, and data packets are identified.
- Business requirements determine how to define traffic into traffic classes, from highest priority to lowest priority.
- A QoS policy is a network-wide definition of the specific levels of QoS assigned to classes of network traffic.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3—1-12

Implementing QoS

Overview

Cisco recommends using either the Modular QoS command-line interface (MQC) or Cisco AutoQoS for implementing quality of service (QoS) in a network. The MQC offers a highly modular way to fine-tune a network. AutoQoS offers an automated method for almost instantly incorporating consistent voice QoS in a network of routers and switches. Network administrators and architects benefit from using the CiscoWorks QoS Policy Manager (QPM). CiscoWorks QPM provides centralized QoS design, administration, and traffic monitoring that scales to large QoS deployments. This lesson explores in detail the four methods for implementing and managing QoS.

Objectives

Upon completing this lesson, you will be able to list and describe methods for implementing QoS. This ability includes being able to meet these objectives:

- List four methods for implementing and managing a QoS policy: CLI, MQC, AutoQoS, and QPM
- Describe the CLI method for implementing QoS on a network
- Describe the MQC method for implementing QoS on a network
- Describe the use of Cisco AutoQoS for implementing QoS on a network
- Explain the advantages and disadvantages of using each of the methods of implementing QoS on a network
- Describe the use of CiscoWorks QPM for managing QoS on a network
- Describe the key MIBs that are used in monitoring QoS implementations
- Describe key MIBs that are used for managing QoS policy on a network

Methods for Implementing QoS Policy

This topic describes four different methods for implementing and managing a QoS policy.

Methods for Implementing QoS Policy

- CLI
- MQC
- AutoQoS VoIP (voice QoS)
- AutoQoS for the Enterprise (voice, video, and data QoS)
- QPM

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-1.2

Just a few years ago, the only way to implement QoS in a network was by using the command-line interface (CLI) to individually configure QoS policies at each interface. This was a time-consuming, tiresome, and error-prone task that involved cutting and pasting configurations from one interface to another.

Cisco introduced the MQC in order to simplify QoS configuration by making configurations modular. Using MQC, you can configure QoS in a building-block approach using a single module repeatedly to apply policy to multiple interfaces.

Cisco AutoQoS represents innovative technology that simplifies the challenges of network administration by reducing QoS complexity, deployment time, and cost to enterprise networks. Cisco AutoQoS incorporates value-added intelligence in Cisco IOS Software and Cisco Catalyst software to provision and assist in the management of large-scale QoS deployments. AutoQoS is an intelligent macro that enables you to enter one or two simple AutoQoS commands to enable all the appropriate features for the recommended QoS setting for an application on a specific interface.

There are these two versions of AutoQoS:

- **AutoQoS VoIP:** In its initial release, AutoQoS VoIP provided best-practice QoS configuration for VoIP on both Cisco switches and routers. This was accomplished by entering one global or interface command. Depending on the platform, the AutoQoS macro would then generate commands into the recommended VoIP QoS configurations, along with class maps and policy maps, and apply those to a router interface or switch port. AutoQoS is available on both LAN and WAN Cisco Catalyst switches and Cisco IOS routers. In its initial version, AutoQoS applied to VoIP deployments, until the release of a new feature in AutoQoS, which is only applicable to the router platform and includes a traffic discovery mechanism that detects not only VoIP traffic but video and data traffic as well.
- **AutoQoS for the Enterprise:** AutoQoS for the Enterprise relies on network-based application recognition (NBAR) to gather statistics and detect ten traffic types, resulting in the provisioning of class maps and policy maps for these traffic types. This feature deploys best-practice QoS policies for voice, video, and data traffic, and is introduced in Cisco IOS Software Release 12.3(7)T. These ten traffic types are detected as traffic crosses the WAN interfaces.

AutoQoS for the Enterprise, combined with the AutoQoS VoIP command, allows a novice network administrator to administer complex detailed QoS policies throughout the enterprise network. AutoQoS for the Enterprise is only for Cisco IOS router platforms. The VoIP feature for Cisco Catalyst switches does not change.

There are some major differences between AutoQoS VoIP and AutoQoS for the Enterprise. AutoQoS VoIP does not detect traffic types nor does it use NBAR. AutoQoS VoIP only creates QoS policy to provide priority of voice traffic. AutoQoS for the Enterprise, on the other hand, uses a discovery mechanism or traffic data collection process that uses NBAR. The AutoQoS VoIP macros use the NBAR statistics to create QoS policies.

You can more easily provision and manage successful QoS deployments by using Cisco AutoQoS together with CiscoWorks QPM. Cisco AutoQoS provides QoS provisioning for individual routers and switches, simplifying deployment and reducing human error. CiscoWorks QPM provides centralized QoS design, administration, and traffic monitoring that scales to large QoS deployments.

Legacy CLI

This topic describes the CLI method for implementing QoS.

Implementing QoS with CLI

- Traditional method
- Nonmodular
- Cannot separate traffic classification from policy definitions
- Used to augment or fine-tune newer AutoQoS method

```
interface Multilink1
 ip address 10.1.61.1 255.255.255.0
 ip tcp header-compression iphc-format
 load-interval 30
 custom-queue-list 1
 ppp multilink
 ppp multilink fragment-delay 10
 ppp multilink interleave
 multilink-group 1
 ip rtp header-compression iphc-format
!
```

© 2003 Cisco Systems, Inc. All rights reserved.

QoS v2.3-1.3

At one time, CLI was the only way to implement QoS in a network. It was a painstaking task, involving copying one interface configuration and then pasting it into other interface configurations. CLI took a lot of time and patience.

The original CLI method was nonmodular—there was no way to separate the classification of traffic from the actual definition of policy. You had to do both on every interface. The figure illustrates an example of the complex configuration tasks involved in using CLI.

While CLI is not recommended for implementing QoS policy, it is still used to fine-tune QoS implementations that have been generated using the Cisco AutoQoS macro.

Modular QoS CLI

This topic describes the MQC method for implementing QoS.

Implementing QoS with MQC

- A command syntax for configuring QoS policy
- Reduces configuration steps and time
- Configure policy, not “raw” per-interface commands
- Uniform CLI across major Cisco IOS platforms
- Uniform CLI structure for all QoS features
- Separates classification engine from the policy

```
class-map VoIP-RTP
match access-group 100
class-map VoIP-Control
match access-group 101
!
policy-map QoS-Policy
class VoIP-RTP
priority 100
class VoIP-Control
bandwidth 8
class class-default
fair-queue
!
interface serial 0/0
service-policy output QoS-Policy
!
access-list 100 permit ip any any
precedence 5
access-list 100 permit ip any any dscp ef
access-list 101 permit tcp any host
10.1.10.20 range 2000 2002
access-list 101 permit tcp any host
10.1.10.20 range 11000 11999
```

The MQC is a CLI structure that allows you to create traffic policies and then attach these policies to interfaces. A traffic policy contains one or more traffic classes and one or more QoS features. A traffic class is used to classify traffic; the QoS features in the traffic policy determine how to treat the classified traffic.

The MQC offers significant advantages over the legacy CLI method for implementing QoS. By using MQC, you can significantly reduce the time and effort it takes to configure QoS on a complex network. Rather than configuring “raw” CLI commands interface by interface, you develop a uniform set of traffic classes and QoS policies that can be applied on interfaces.

The use of the MQC allows the separation of traffic classification from the definition of QoS policy. This enables easier initial QoS implementation and maintenance as new traffic classes emerge and QoS policies for the network evolve.

AutoQoS VoIP and Enterprise

This topic describes the use of Cisco AutoQoS to implement QoS in a network.

Implementing QoS with AutoQoS

- AutoQoS VoIP supported both in the LAN and WAN environments
- AutoQoS Enterprise supported on WAN interfaces
- Routers can deploy Enterprise QoS policy treatment for voice, video, and data traffic
- Switches can deploy QoS policy treatments for voice by a single command

The [trust] option indicates that the DSCP or CoS markings of a packet are relied upon for classification.

WAN

AutoQoS VoIP:

```
auto qos voip [trust]
```

Port

WAN

AutoQoS Enterprise:

```
auto discovery qos [trust]  
auto qos
```

© 2009 Cisco Systems, Inc. All rights reserved. QoS v2.3-1.6

Depending on the platform, using Cisco AutoQoS VoIP or AutoQoS for the Enterprise, you can implement the QoS features that support VoIP traffic and data traffic without an in-depth knowledge of these underlying technologies:

- PPP
- Frame Relay
- ATM
- Service policies
- Link efficiency mechanisms, such as link fragmentation and interleaving (LFI)

The deployment of AutoQoS simplifies QoS implementation, speeds up the provisioning of QoS technology over a Cisco network, reduces human error, and lowers training costs.

The AutoQoS for the Enterprise feature, introduced in Cisco IOS Software Release 12.3(7)T, provides another option for configuring AutoQoS on routers. This new feature allows you to configure AutoQoS to discover data traffic across a router WAN interface before applying AutoQoS VoIP. While previously you could apply AutoQoS VoIP on a WAN interface, this new automated feature allows you to discover data traffic and build the QoS policies based on data gathered during the discovery phase. Configuring switches to support AutoQoS VoIP has not changed. This new feature applies only to router WAN interfaces, and the phases are described as follows:

- Phase 1 is the autodiscovery (data collection) phase and uses NBAR to detect network applications as they arrive at an interface, collect data from the offered traffic, and perform statistical analysis. The data collected is a representative sampling of the volume and type of voice, video, and data on the customer network. The amount of time devoted to data collection varies from network to network.
- Phase 2 generates templates from the data collected during the autodiscovery phase and installs the templates on the interface. These templates are used as the basis for creating the class maps and policy maps for your network. After the class maps and policy maps are created, they are installed on the interface. During this phase, the AutoQoS for the Enterprise also assigns the appropriate bandwidth amounts and sets the appropriate scheduling parameters for the network traffic.
- You can use CiscoWorks QPM in conjunction with the AutoQoS for the Enterprise feature to provide a centralized, web-based tool to cost-effectively manage and monitor network-wide QoS policies. The AutoQoS VoIP feature, together with CiscoWorks QPM, eases QoS implementation, provisioning, and management.

Note	Cisco AutoQoS was introduced in Cisco IOS Software Release 12.2(15)T. AutoQoS discovery was introduced in Cisco IOS Software Release 12.3(7)T.

QoS Implementation Methods Compared

This topic describes the advantages and disadvantages of using each of the methods of implementing QoS on a network.

Comparing Methods for Implementing QoS				
	CLI	MQC	AutoQoS VoIP	AutoQoS Enterprise
Ease of Use	Poor	Easier	Simple	Simple
Ability to Fine-Tune	OK	Very Good	Very Good	Very Good
Time to Deploy	Longest	Average	Shortest	Shortest
Modularity	Poor	Excellent	Excellent	Excellent

© 2003 Cisco Systems, Inc. All rights reserved. QoS v2.3-1.6

Cisco recommends the use of MQC and AutoQoS VoIP when deploying voice over the LAN, and AutoQoS for the Enterprise on router WAN interfaces.

While MQC is much easier to use than CLI, AutoQoS VoIP and AutoQoS for the Enterprise can simplify the configuration of QoS. As a result, you can accomplish the fastest implementation with AutoQoS.

MQC offers excellent modularity and the ability to fine-tune complex networks. AutoQoS offers the fastest way to implement QoS, but has limited fine-tuning capabilities. When an AutoQoS configuration has been generated, you must use CLI commands to fine-tune an AutoQoS configuration if necessary. (On most networks, fine-tuning will not be necessary for AutoQoS.)

QoS Policy Manager

This topic describes the use of CiscoWorks QPM to manage QoS on a network.

QoS Policy Manager

Suite of management functions that allow network administrators to fully leverage the Cisco intelligent IP infrastructure, enable network-wide QoS, and obtain precise, easy-to-understand QoS information with monitoring and reporting.

Provisioning



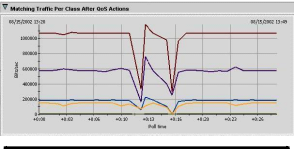
- Recommendations via wizards, templates
- Verification
- Customization

Baseline Analysis



- Analysis of existing network traffic to help you determine the types of policies from which your network might benefit

QoS Analysis



- Historical and real-time analyses of how your QoS policies are affecting network traffic
- Historical and real-time QoS reports that display detailed QoS statistics

© 2009 Cisco Systems, Inc. All rights reserved. QoS v2.3-1.7

CiscoWorks QPM provides a scalable platform for defining, applying, and monitoring QoS policy on a system-wide basis for Cisco devices, including routers and switches.

CiscoWorks QPM enables the baselining of profile network traffic, creates QoS policies at an abstract level, controls the deployment of policies, and monitors QoS to verify intended results. As a centralized tool, CiscoWorks QPM is used to monitor and provision QoS for groups of interfaces and devices.

CiscoWorks QPM provides a web-based intuitive user interface to define QoS policies and translates those policies into the device CLI commands.

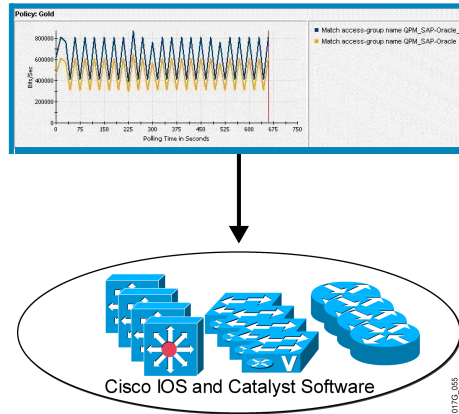
CiscoWorks QPM allows for analyzing of traffic throughput by application or service class. This analysis leverages that information to configure QoS policies to differentiate traffic and define the QoS functions that are applied to each type of traffic flow.

By simplifying QoS policy definition and deployment, CiscoWorks QPM makes it easier to create and manage end-to-end differentiated services in a network, thus making more efficient and economical use of existing network resources. For example, policies can be deployed that ensure that mission-critical applications always get the bandwidth required to run the business.

Cisco AutoQoS with CiscoWorks QPM

- Use AutoQoS Enterprise to configure routers
- Use AutoQoS VoIP on switches

Use QPM to manage network-wide QoS for multiple devices

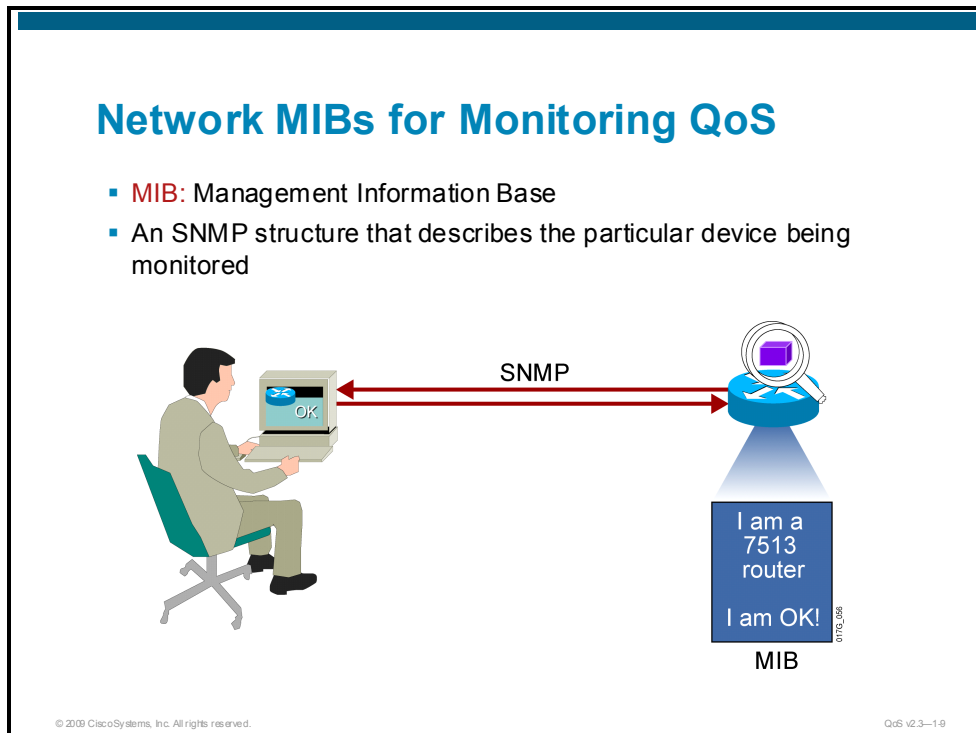


© 2009 Cisco Systems, Inc. All rights reserved.

Customers can more easily provision and manage successful QoS deployments using Cisco AutoQoS together with CiscoWorks QPM. Cisco AutoQoS provides QoS provisioning for individual routers and switches, simplifying deployment and reducing human error. CiscoWorks QPM provides centralized QoS design, administration, and traffic monitoring that scales to large QoS deployments.

Network MIBs for Monitoring QoS

This topic describes the key MIBs that are used to monitor QoS implementations.



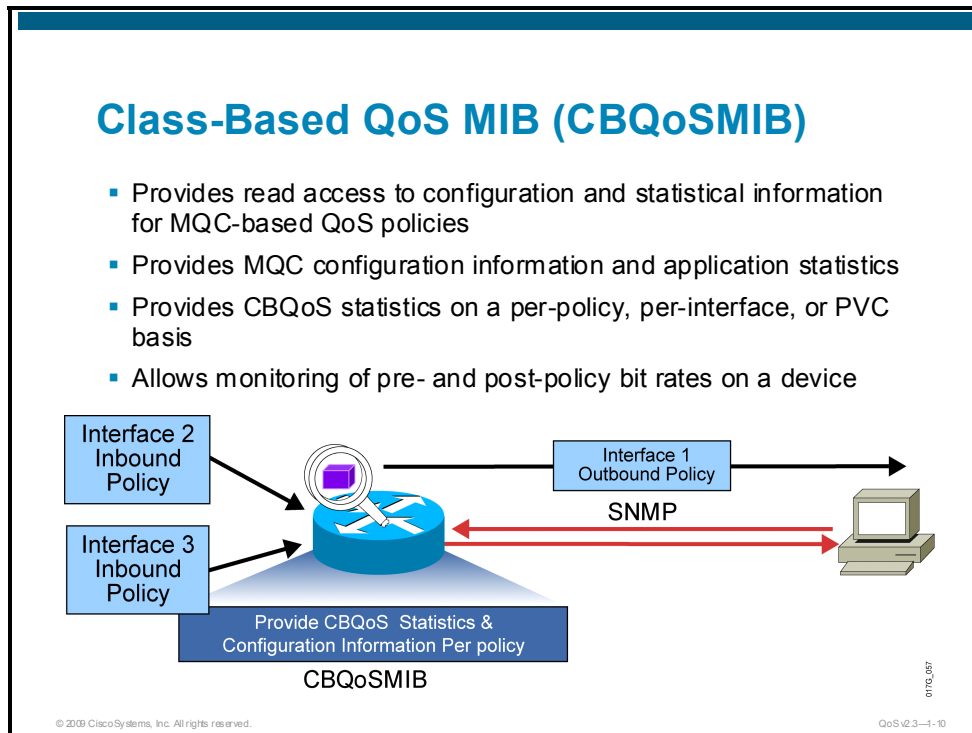
A MIB is a Simple Network Management Protocol (SNMP) structure that describes the particular device being monitored. Cisco provides many standards-based MIBs for use in monitoring the status of devices on a network.

Advanced network management products, such as CiscoWorks QPM, use these MIBs to generate statistics about the performance of the network. Specialized QoS MIBs enable CiscoWorks QPM to graphically display key QoS information to aid in the management of QoS policies on the network.

Note For more information on Cisco MIBs, refer to Network Management Software at this URL:
<http://www.cisco.com/public/sw-center/sw-netmgmt.shtml>

MIBs for Managing QoS

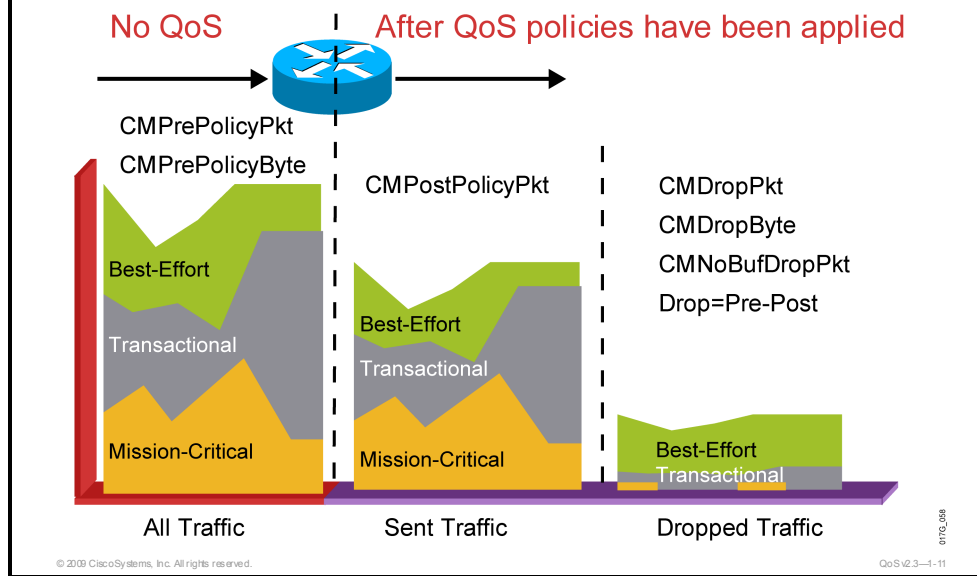
This topic describes the key MIBs that are used for managing QoS policy on a network.



The Class-Based QoS MIB (CBQoS MIB) provides read access to QoS configurations. This MIB also provides QoS statistics information based on the MQC, including information regarding class map and policy map parameters.

This CBQoS MIB actually contains two MIBs: Cisco Class-Based QoS MIB and Cisco Class-Based QoS Capability MIB.

QPM: Monitoring and Reporting with CBQoS MIB



CiscoWorks QPM uses the information collected in the class-based MIB to build a number of reports showing the effect of QoS policies on the network.

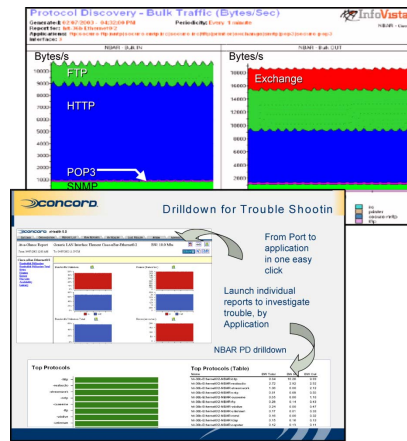
These reports can graphically illustrate the overall input traffic flow divided by traffic class, the traffic that was actually sent, and the traffic that was dropped because of QoS policy enforcement.

In the Reports tab for CiscoWorks QPM, under Historical Reports, you can create graphs such as the following:

- **Matching Traffic Per Class *Prior to QoS* Actions:** Graphs that display the traffic that matched each policy group filter before any policy actions were performed
- **Matching Traffic Per Class *After QoS* Actions:** Graphs that display the traffic that matched each policy group filter and was transmitted (not dropped) by the configured QoS policies
- **Matching Traffic Per Class *Discarded by QoS Drop* Actions:** Graphs that display the traffic that matched each policy group filter and was dropped (not transmitted) by QoS policy drop actions

Cisco NBAR Protocol Discovery MIB

- NBAR protocol discovery statistics only available on the configured device
- Provides ability to retrieve statistics via SNMP into a central performance monitoring system



Another important MIB that is used for monitoring QoS is the Cisco NBAR Protocol Discovery MIB. Using the information collected by this MIB, it is possible to collect detailed protocol and application-level network statistics.

Note NBAR protocol discovery MIB is discussed further in the module “Classification and Marking,” where NBAR is explained.

Summary

This topic summarizes the key points discussed in this lesson.

Summary

- There are four different methods for implementing QoS: CLI, MQC, AutoQoS, and CiscoWorks QPM.
- The original CLI method for implementing QoS is a nonmodular and time-consuming method of implementing QoS.
- MQC is a CLI structure that allows you to create traffic policies and then attach these policies to interfaces.
- AutoQoS is an intelligent macro that enables you to enter one or two simple commands to automatically implement a set of QoS policies on a router or switch.
- The MQC offers excellent modularity and the ability to fine-tune complex networks; AutoQoS offers the fastest way to implement QoS, but has limited fine-tuning capabilities.
- QPM enables the baselining of profile network traffic, creates QoS policies at an abstract level, controls the deployment of policies, and monitors QoS to verify intended results.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3—1-13

Summary (Cont.)

- Specialized QoS MIBs enable CiscoWorks QPM to graphically display key QoS information to aid in the management of QoS policies on the network.
- The Class-Based QoS MIB (CBQoS MIB) provides read access to QoS configurations and provides QoS statistics information based on the MQC.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3—1-14

Module Summary

This topic summarizes the key points discussed in this module.

Module Summary

- The problems that can lead to poor QoS for applications running on a converged network include lack of bandwidth, excessive delay, jitter, and packet loss.
- QoS is the ability of the network to provide better or “special” service to select users and applications.
- The key steps to implement QoS on a converged network include:
 - Identifying the traffic types and their requirements
 - Classifying the traffic
 - Defining and implementing QoS policies for each traffic class

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3–1.1

Module Summary (Cont.)

- Voice, video, and data have very different requirements and must be defined into traffic classes.
- Different methods to implement a QoS policy on Cisco devices include the following:
 - Use existing CLI configuration mode
 - Use MQC-based QoS configuration
 - Use automated methods such as AutoQoS and CiscoWorks QPM

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3–1.2

Converged IP networks must provide secure, predictable, measurable, and sometimes guaranteed services. QoS provides network administrators and architects with a set of techniques used to manage network resources. Cisco IOS Software provides a rich set of tools and an automated approach for implementing QoS, such as AutoQoS.

References

For additional information, refer to these resources:

- *Implementing Quality of Service* at this URL:
http://www.cisco.com/en/US/partner/tech/tk543/tk757/technologies_white_paper09186a008017f93b.shtml
- To learn more about QoS, refer to Cisco IOS Quality of Service (QoS) at this URL:
http://www.cisco.com/en/US/technologies/tk389/tk813/technologies_white_paper0900aecd802b68b1.html
- For more information on QPM, refer to *Introduction to QPM* at this URL:
http://www.cisco.com/en/US/products/sw/cscowork/ps2064/products_user_guide_chapter09186a00800e0a00.html
- For more information on Cisco MIBs, refer to Network Management Software at this URL:
<http://www.cisco.com/public/sw-center/sw-netmgmt.shtml>

Module Self-Check

Use the questions here to review what you learned in this module. The correct answers and solutions are found in the Module Self-Check Answer Key.

- Q1) Which of these terms is used to describe the time that it takes to actually transmit a packet on a link (“put bits on the wire”)? (Source: Understanding the Need for QoS)
- A) encoding delay
 - B) processing delay
 - C) serialization delay
 - D) transmission delay
- Q2) What is the “best” solution for reducing delay on a link? (Source: Understanding the Need for QoS)
- A) compress data and headers
 - B) drop low-priority packets early
 - C) increase the bandwidth of the link
 - D) incorporate advanced queuing technologies
- Q3) Which three of these are characteristics of converged network traffic? (Choose three.) (Source: Understanding the Need for QoS)
- A) constant small-packet flow
 - B) time-sensitive packets
 - C) intolerance of brief outages
 - D) bursty small packet flow
- Q4) How much one-way delay can a voice packet tolerate? (Source: Understanding QoS)
- A) 15 ms
 - B) 150 ms
 - C) 300 ms
 - D) 200 ms
- Q5) Which transport layer protocol is used for voice traffic? (Source: Understanding QoS)
- A) UDP
 - B) TCP
 - C) XNS
 - D) HTTP
- Q6) Which three of these represent components in the definition of a QoS policy? (Choose three.) (Source: Understanding QoS)
- A) user-validated
 - B) network-wide
 - C) specific levels of quality of service
 - D) different classes of network traffic
- Q7) How are QoS implementations that are generated using AutoQoS fine-tuned? (Source: Implementing QoS)
- A) command-line interface
 - B) Modular QoS CLI
 - C) QoS AutoTune
 - D) QoS Policy Manager

- Q8) Which three of these are advantages of using MQC? (Choose three.) (Source: Implementing QoS)
- A) reduction in time to configure a complex policy
 - B) ability to apply one policy to multiple interfaces
 - C) separation of classification from policy definition
 - D) automatic generation of CLI commands from MQC macros
- Q9) Which QoS implementation method has the quickest implementation time for simple networks? (Source: Implementing QoS)
- A) CLI
 - B) MQC
 - C) AutoQoS
 - D) AutoTuner
- Q10) Which two of these MIBs are specifically designed for managing QoS in a network? (Choose two.) (Source: Implementing QoS)
- A) Modular QoS MIB
 - B) Class-Based QoS MIB
 - C) QoS Policy Manager MIB
 - D) Cisco NBAR Protocol Discovery MIB

Module Self-Check Answer Key

Q1)	C
Q2)	C
Q3)	A, B, C
Q4)	B
Q5)	A
Q6)	B, C, D
Q7)	A
Q8)	A, B, C
Q9)	C
Q10)	B, D

The Building Blocks of QoS

Overview

Quality of service (QoS) and its implementations in a converged network are complex and create many challenges for network administrators and architects. Evolving technology introduces different approaches to providing QoS to network applications. Many QoS building blocks or features operate at different parts of a network to create an end-to-end QoS system. Managing how these building blocks are assembled and how different QoS features are used can be a difficult task. In response to these difficulties, three different implementation models for QoS have been developed.

This module discusses the different implementation models of QoS and describes how the different building blocks of QoS integrate into each model. This module also discusses the different QoS features and where those features are typically implemented within a network.

Module Objectives

Upon completing this module, you will be able to identify and describe different models used for ensuring QoS in a network, and explain key QoS mechanisms that are used to implement the models. This ability includes being able to meet these objectives:

- Correctly match a list of QoS actions to one or more of the three models for implementing QoS on a network
- Describe the Integrated Services model and explain how this model can be used to implement QoS in a network
- Describe the Differentiated Services model and explain how this model can be used to implement QoS in a network
- Correctly match a list of QoS actions to mechanisms for implementing DiffServ QoS and identify where in a network the different QoS mechanisms are commonly used

Identifying Models for Implementing QoS

Overview

Three different models exist for implementing quality of service (QoS) on a network. The best-effort model was designed for best-effort, no-guarantee delivery of packets and is still the predominant model in the Internet today. The Integrated Services (IntServ) model was introduced to supplement the best-effort delivery by setting aside some bandwidth for applications that require bandwidth and delay guarantees. IntServ expects applications to signal their requirements to the network. The Differentiated Services (DiffServ) model was added to provide greater scalability in providing QoS to IP packets. The main difference between the IntServ and DiffServ models is that in DiffServ, the network recognizes packets (no signaling is needed) and provides the appropriate services to the packets. IP networks today can use all three models at the same time.

Objectives

Upon completing this lesson, you will be able to correctly match QoS actions to one or more models for implementing QoS on a network. This ability includes being able to meet these objectives:

- Describe the types of models for providing QoS on a network
- Describe the key features of the best-effort model for QoS
- Describe the key features of IntServ for QoS
- Describe the key features of DiffServ for QoS

QoS Models

This topic describes the types of models for providing QoS on a network.

Three Models for Quality of Service

- **Best-Effort:** No QoS is applied to packets.
- **IntServ:** Applications signal to the network that they require special QoS.
- **DiffServ:** The network recognizes classes that require special QoS.

© 2003 Cisco Systems, Inc. All rights reserved.

QoS v2.3-2.2

The following three models exist for implementing QoS in a network:

- **Best-effort:** With the best-effort model, QoS is not applied to packets. If it is not important when or how packets arrive, the best-effort model is appropriate.
- **IntServ:** IntServ can provide very high QoS to IP packets. Essentially, applications signal to the network that they will require special QoS for a period of time and that bandwidth is reserved. With IntServ, packet delivery is guaranteed. However, the use of IntServ can severely limit the scalability of a network.
- **DiffServ:** DiffServ provides the greatest scalability and flexibility in implementing QoS in a network. Network devices recognize traffic classes and provide different levels of QoS to different traffic classes.

Best-Effort Model

This topic describes the key features of the best-effort model for QoS.

Best-Effort Model

- Internet initially based on a best-effort packet delivery service
- The default mode for all traffic
- No differentiation between types of traffic
- Like using standard mail



It will get there when it gets there.

© 2009 Cisco Systems, Inc. All rights reserved. QoS v2.3-2.3

The Internet was designed for best-effort, no-guarantee delivery of packets. This behavior is still predominant on the Internet today.

If QoS policies are not implemented, traffic is forwarded using the best-effort model. All network packets are treated exactly the same—an emergency voice message is treated exactly like a digital photograph attached to an email. Without the implementation of QoS, the network cannot tell the difference between packets and, as a result, cannot treat packets preferentially.

When you drop a letter in standard postal mail, you are using a best-effort model. Your letter will be treated exactly the same as every other letter; it will get there when it gets there. With the best-effort model, the letter may actually never arrive and, unless you have a separate notification arrangement with the letter recipient, you may never know if the letter does not arrive.

Best-Effort Model (Cont.)

- **Benefits:**
 - Highly scalable
 - No special mechanisms required
- **Drawbacks:**
 - No service guarantees
 - No service differentiation

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-2.4

The best-effort model has these significant benefits:

- The best-effort model has nearly unlimited scalability. The only way to reach scalability limits is to reach bandwidth limits, in which case all traffic becomes equally delayed.
- You do not need to employ special QoS mechanisms to use the best-effort model. It is the easiest and quickest model to deploy.

The best-effort model also has these drawbacks:

- Nothing is guaranteed. Packets will arrive whenever they can, in any order possible, if they arrive at all.
- Packets are not given preferential treatment. Critical data is treated the same as casual email.

IntServ Model

This topic describes the key features of the IntServ model for QoS.

IntServ Model

- Some applications have special bandwidth or delay requirements or both
- IntServ guarantees a predictable behavior of the network for these applications
- Guaranteed delivery: no other traffic can use reserved bandwidth
- Like having your own private courier plane



It will be there by 10:30 a.m.

Some applications, such as high-resolution video, require consistent, dedicated bandwidth to provide sufficient quality for viewers. IntServ was introduced to guarantee predictable network behavior for these applications.

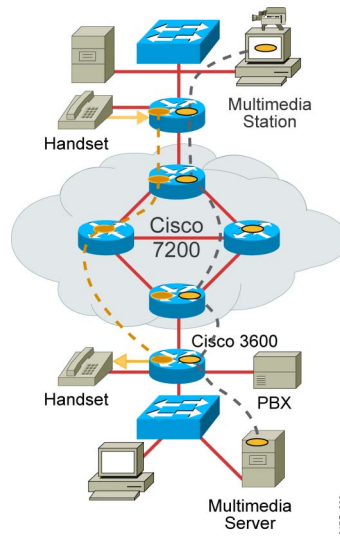
Because IntServ reserves bandwidth throughout a network, no other traffic can use the reserved bandwidth. Bandwidth that is unused, but reserved, is wasted.

IntServ is similar to a concept known as “hard QoS.” With hard QoS, traffic characteristics such as bandwidth, delay, and packet-loss rates are guaranteed end-to-end. This guarantee ensures both predictable and guaranteed service levels for mission-critical applications. There will be no impact on traffic when guarantees are made, regardless of additional network traffic. Hard QoS is accomplished by negotiating specific QoS requirements upon establishment of a connection and by using Call Admission Control (CAC) to ensure that no new traffic will violate the guarantee. Such guarantees require an end-to-end QoS approach with both complexity and scalability limitations. Large network environments that contain heavy traffic loads will be extremely challenged to track QoS guarantees for hundreds of thousands of signaled flows.

Using IntServ is like having a private courier airplane or truck dedicated to the delivery of your traffic. This model ensures quality and delivery, is expensive, and is not scalable.

IntServ Model (Cont.)

- Provides multiple service levels
- Requests specific kind of service from the network before sending data
- Uses RSVP to reserve network resources
- Uses intelligent queuing mechanisms
- End-to-end



© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-2.6

IntServ is a multiple-service model that can accommodate multiple QoS requirements. IntServ inherits the connection-oriented approach from telephony network design. Every individual communication must explicitly specify its traffic descriptor and requested resources to the network. The edge router performs admission control to ensure that available resources are sufficient in the network. The IntServ standard assumes that routers along a path set and maintain the state for each individual communication.

The role of Resource Reservation Protocol (RSVP) in the Cisco QoS architecture is to provide resource admission control for VoIP networks. If resources are available, RSVP accepts a reservation and installs a traffic classifier in the QoS forwarding path. The traffic classifier tells the QoS forwarding path how to classify packets from a particular flow and what forwarding treatment to provide.

In the IntServ model, the application requests a specific kind of service from the network before sending data. The application informs the network of its traffic profile and requests a particular kind of service that can encompass its bandwidth and delay requirements. The application is expected to send data only *after* it gets a confirmation from the network. The application is also expected to send data that lies within its described traffic profile.

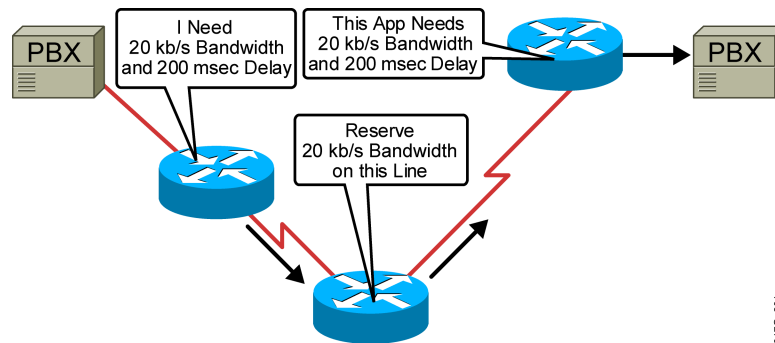
The network performs admission control based on information from the application and available network resources. The network commits to meeting the QoS requirements of the application as long as the traffic remains within the profile specifications. The network fulfills its commitment by maintaining the per-flow state, and then performing packet classification, policing, and intelligent queuing based on that state.

The QoS feature set in Cisco IOS Software includes these features that provide controlled-load service:

- RSVP can be used by applications to signal their QoS requirements to the router.
- Intelligent queuing mechanisms can be used with RSVP to provide these QoS service levels:
 - **Guaranteed-rate:** Guaranteed-rate service level allows applications to reserve bandwidth to meet their requirements. For example, a VoIP application can reserve 32 Mb/s end-to-end using this type of service. Cisco IOS QoS uses low latency queuing (LLQ) with RSVP to provide guaranteed-rate type of service.
 - **Controlled-load:** Controlled-load service level allows applications to have low delay and high throughput, even during times of congestion. For example, adaptive real-time applications, such as the playback of a recorded conference, can use this service. Cisco IOS QoS uses RSVP with weighted random early detection (WRED) to provide controlled-load type of service.

IntServ Model (Cont.)

- RSVP QoS services
 - Guaranteed-rate service
 - Controlled-load service
- RSVP provides policy to QoS mechanisms



RSVP is an IP service that allows end systems or hosts on either side of a router network to establish a reserved-bandwidth path between them to predetermine and ensure QoS for their data transmission. RSVP is currently the only standard signaling protocol designed to guarantee network bandwidth from end to end for IP networks.

RSVP is an Internet Engineering Task Force (IETF) standard (RFC 2205) protocol for allowing an application to dynamically reserve network bandwidth. RSVP enables applications to request a specific QoS for a data flow (shown in the figure). Cisco implementation also allows RSVP to be initiated within the network, using configured proxy RSVP. Network managers can take advantage of RSVP benefits in the network, even for non-RSVP-enabled applications and hosts.

Hosts and routers use RSVP to deliver QoS requests to the routers along the paths of the data stream. Hosts and routers also use RSVP to maintain the router and host state to provide the requested service, usually bandwidth and latency. RSVP uses a mean data rate; that is, the largest amount of data the router will keep in queue and the minimum QoS used to determine bandwidth reservation.

LLQ or WRED act as the workhorses for RSVP, setting up the packet classification and scheduling that is required for the reserved flows. Using LLQ, RSVP can deliver an IntServ-guaranteed service. Using WRED, RSVP can deliver a controlled-load service. RSVP can be deployed in existing networks with a software upgrade.

IntServ Model (Cont.)

- **Benefits:**
 - Explicit resource admission control (end to end)
 - Per-request policy admission control (authorization object, policy object)
 - Signaling of dynamic port numbers (for example, H.323)
- **Drawbacks:**
 - Continuous signaling because of stateful architecture
 - Flow-based approach not scalable to large implementations such as the public Internet (can be made more scalable when combined with elements of the DiffServ model)

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-2.8

The main benefits of IntServ and RSVP are as follows:

- RSVP signals QoS requests per individual flow. The network can then provide guarantees to these individual flows. The problem with this is that IntServ does not scale to large networks because of the large number of concurrent RSVP flows.
- RSVP informs network devices of flow parameters (IP addresses and port numbers). Some applications use dynamic port numbers, which can be difficult for network devices to recognize. Network-Based Application Recognition (NBAR) is a mechanism that has been introduced to supplement RSVP for applications that use dynamic port numbers but do not use RSVP.
- IntServ supports admission control, which allows a network to reject (or downgrade) new RSVP sessions if one of the interfaces in the path has reached the limit (that is, all reservable bandwidth is booked).

The main drawbacks of IntServ and RSVP are as follows:

- There is continuous signaling because of the stateful RSVP operation.
- RSVP is not scalable to large networks where per-flow guarantees would have to be made to thousands of concurrent RSVP flows.

DiffServ Model

This topic describes the key features of the DiffServ model for QoS.

DiffServ Model

- Network traffic identified by class
- Network QoS policy enforces differentiated treatment of traffic classes
- You choose level of service for each traffic class
- Like using a package delivery service



Do you want overnight delivery?

Do you want two-day air delivery?

Do you want three- to seven-day ground delivery?

© 2009 Cisco Systems, Inc. All rights reserved. QoS v2.3-2.9

DiffServ was designed to overcome the limitations of both the best-effort and IntServ models. DiffServ can provide an “almost guaranteed” QoS, while still being cost-effective and scalable.

DiffServ is similar to a concept known as “soft QoS.” With soft QoS, QoS mechanisms are used without prior signaling. In addition, QoS characteristics (bandwidth and delay, for example), are managed on a hop-by-hop basis by policies that are established independently at each intermediate device in the network. The soft QoS approach is not considered an end-to-end QoS strategy because end-to-end guarantees cannot be enforced. However, soft QoS is a more scalable approach to implementing QoS than hard QoS, because many (hundreds or potentially thousands) of applications can be mapped into a small set of classes upon which similar sets of QoS behaviors are implemented. Although QoS mechanisms in this approach are enforced and applied on a hop-by-hop basis, uniformly applying global meaning to each traffic class provides both flexibility and scalability.

With DiffServ, network traffic is divided into classes based on business requirements. Each of the classes can then be assigned a different level of service. As the packets traverse a network, each of the network devices identifies the packet class and services the packet according to that class. You can choose many levels of service with DiffServ. For example, voice traffic from IP phones is usually given preferential treatment over all other application traffic. Email is generally given best-effort service. Nonbusiness traffic can either be given very poor service or blocked entirely.

DiffServ works like a package delivery service. You request (and pay for) a level of service when you send your package. Throughout the package network, the level of service is recognized and your package is given either preferential or normal service, depending on what you requested.

DiffServ Model (Cont.)

- **Benefits:**
 - Highly scalable
 - Many levels of quality possible
- **Drawbacks:**
 - No absolute service guarantee
 - Complex mechanisms

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3--2-10

DiffServ has these key benefits:

- It is highly scalable.
- It provides many different levels of quality.

DiffServ also has these drawbacks:

- No absolute guarantee of service quality can be made.
- It requires a set of complex mechanisms to work in concert throughout the network.

Summary

This topic summarizes the key points discussed in this lesson.

Summary

- There are three different models for providing QoS: Best-effort, IntServ, and DiffServ.
- While the best-effort model is highly scalable, it has no provision for differentiating among types of network traffic and, as a result, does not provide QoS.
- The Integrated Services model offers absolute QoS guarantees by explicitly reserving bandwidth, but is not scalable.
- The Differentiated Services model provides the ability to classify network traffic and offer many levels of QoS, while being highly scalable.

Understanding the Integrated Services Model

Overview

The Integrated Services (IntServ) model is an architecture that offers guaranteed quality of service (QoS) end-to-end over the network. The key components of IntServ are Resource Reservation Protocol (RSVP) and Call Admission Control (CAC). This lesson describes the various aspects of the IntServ model and how it uses RSVP.

Objectives

Upon completing this lesson, you will be able to describe IntServ and explain how it can be used to implement QoS in the network. This ability includes being able to meet these objectives:

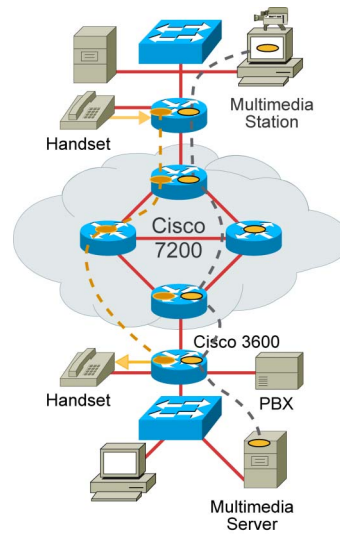
- Explain the key features of the IntServ model for QoS
- Describe the IntServ component, RSVP, and its function
- Explain how bandwidth is used on an interface configured for RSVP
- Explain how to enable RSVP on an interface
- Explain how the IntServ model integrates into enterprise networks

Integrated Services Model

This topic describes the purpose and function of the IntServ model.

IntServ Model

- IntServ provides end-to-end QoS service for applications.
- RSVP mechanism is used to provision end-to-end QoS treatment.
- CAC is used through the network to ensure resource availability.



© 2003 Cisco Systems, Inc. All rights reserved.

QoS v2.3-2.2

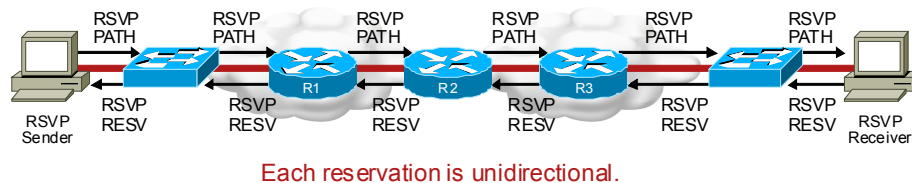
IntServ is an architecture for providing end-to-end QoS. The IntServ solution allows end stations to make requests upon the network. The network participates in this QoS scheme by either reserving or not reserving network resources for the requesting end station. The IntServ architecture provides a way to guarantee network quality levels by specifically reserving services and controlling the load of traffic on devices to provide the guaranteed service requirements.

The IntServ module uses a CAC component and a policy control component to manage requests for reservations. In other words, these components are used by RSVP-enabled routers to determine whether there are enough resources to support the reservation request and whether the host requesting the reservation can be granted a reservation.

IntServ Model (Cont.)

RSVP supports the following messages:

- PATH
- RESV
- Error and Confirmation
- Teardown



The figure illustrates a typical RSVP message being sent and the messages generated by the receiver in the establishment of a reservation. These messages are required in order for a unidirectional reservation to be made. In this example, the sender has successfully requested a reservation to the end host. It is important to note that this is only a one-way reservation.

RSVP supports these messages:

- **Path messages (PATH):** An RSVP path message is sent by each sender along the unicast or multicast routes provided by the routing protocol. A path message is used to store the path state in each node. The path state is used to route reservation-request messages in the reverse direction.
- **Reservation-request messages (RESV):** A reservation-request message is sent by each receiver host toward the senders. This message follows in reverse the routes that the data packets use, all the way to the sender hosts. A reservation-request message must be delivered to the sender hosts so that the hosts can set up appropriate traffic-control parameters for the first hop.
- **Error and confirmation messages:** Reservation-request acknowledgment messages are sent as the result of the appearance of a reservation-confirmation object in a reservation-request message. This acknowledgment message contains a copy of the reservation confirmation. An acknowledgment message is sent to the unicast address of a receiver host, and the address is obtained from the reservation-confirmation object. A reservation-request acknowledgment message is forwarded to the receiver hop-by-hop to accommodate the integrity-check mechanism.
 - Path error messages result from path messages and travel toward senders. Path error messages are routed hop-by-hop using the path state. At each hop, the IP destination address is the unicast address of the previous hop.

- Reservation-request error messages result from reservation-request messages and travel toward the receiver. Reservation-request error messages are routed hop-by-hop using the reservation state. At each hop, the IP destination address is the unicast address of the next-hop node. Information carried in error messages can include the following:
 - Admission failure
 - Bandwidth unavailable
 - Service not supported
 - Bad flow specification
 - Ambiguous path
- **Teardown messages:** RSVP teardown messages remove the path and reservation state without waiting for the cleanup timeout period. Teardown messages can be initiated by an application in an end system (sender or receiver) or a router as the result of state timeout. RSVP supports the following two types of teardown messages:
 - **path-teardown:** Path-teardown messages delete the path state (which deletes the reservation state), travel toward all receivers downstream from the point of initiation, and are routed like path messages.
 - **reservation-request teardown:** Reservation-request teardown messages delete the reservation state, travel toward all matching senders upstream from the point of teardown initiation, and are routed like corresponding reservation-request messages.

IntServ Model (Cont.)

There are three types of QoS services offered by RSVP:

- Best Effort
- Guaranteed Rate
- Controlled Load

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-2.4

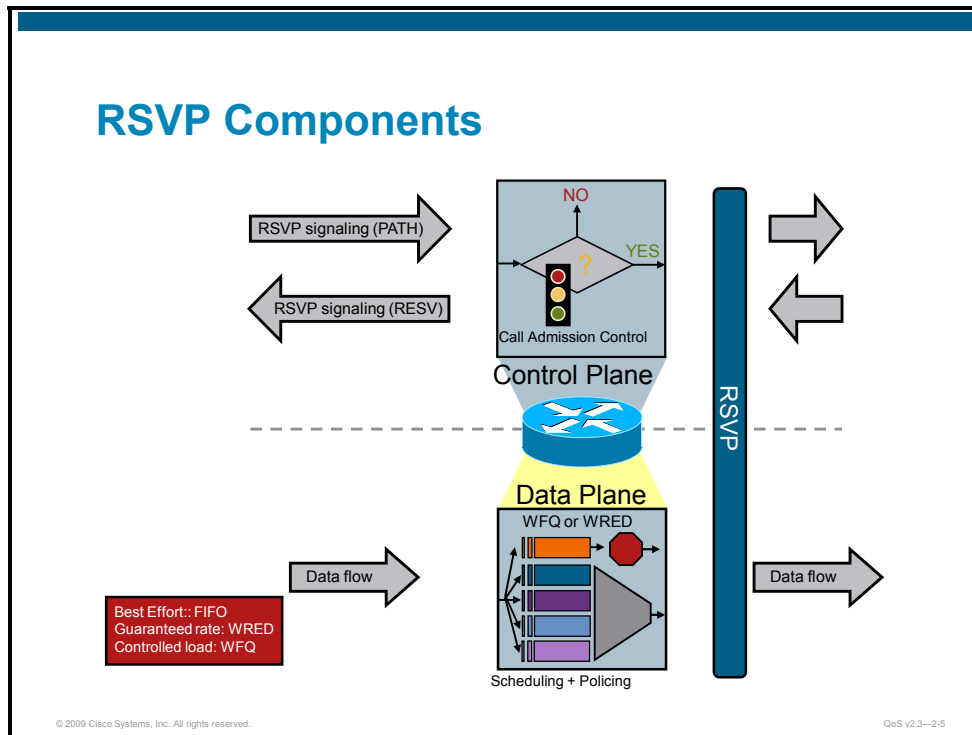
There are three types of QoS services offered by RSVP:

- **Best effort:** There is no QoS at all, just first in, first out.
- **Guaranteed rate:** The RSVP-enabled routers try to guarantee the worst-case delay that will be incurred by the flow when traveling across the network. Guaranteed-rate services compute the delay taken from the PATH messages along the RSVP path of the flow and provide this information to the receiver during the resource reservation request.
- **Controlled Load:** Controlled-load services allow an RSVP session to flow through the network with the least possible interruption from other traffic flows.

If RSVP is configured to prefer a guaranteed rate, weighted random early detection (WRED) is used to ensure QoS. If a controlled load is preferred, weighted fair queuing (WFQ) is used. RSVP is used with DiffServ. When deployed with DiffServ, RSVP is used solely for CAC, and the data flow is then passed through QoS policies such as low-latency queuing (LLQ). When LLQ is used, the data traffic for the RSVP reservation is passed through the low-latency priority queue.

Components of RSVP

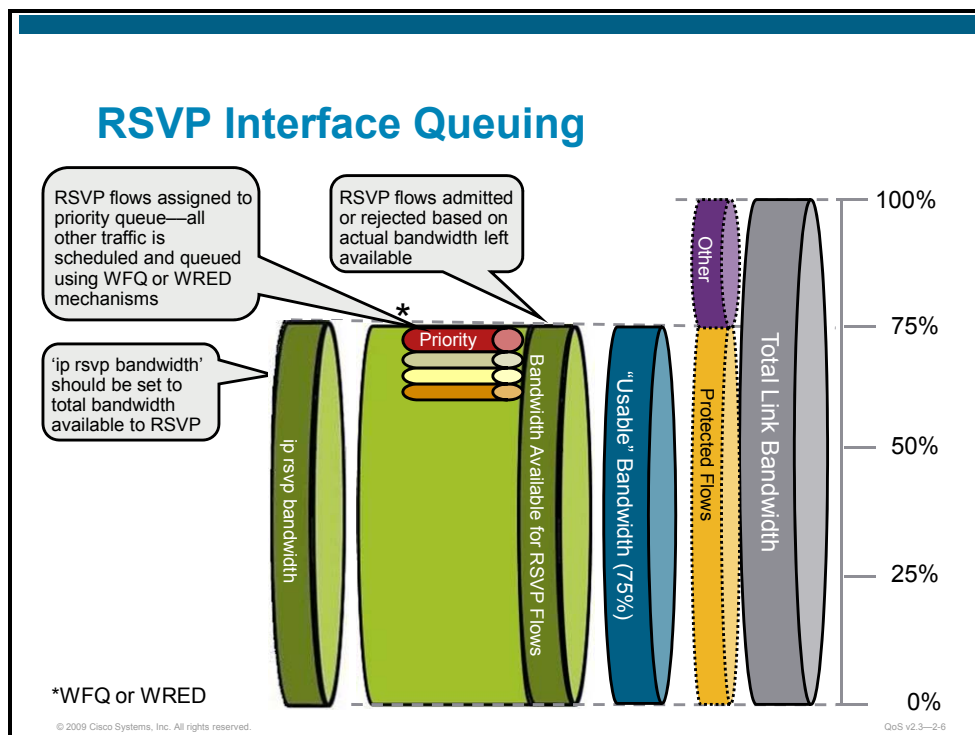
This topic describes the components of RSVP.



In the IntServ model, RSVP involves use of both the control plane and the data plane. In the control plane, RSVP provides a CAC function for reservation request which is based on whether the router has the resources to support the reservation request and whether the host making the request has the right to request the reservation. In the data plane, RSVP data flows are classified and policed based on the traffic descriptions contained in the RSVP messages, and queued to the appropriate queues.

RSVP Interface Bandwidth Queuing

This topic describes how bandwidth is used on an interface configured for RSVP.



The figure outlines how RSVP data flows are allocated when RSVP is configured on an interface. The maximum bandwidth available on any interface is 75 percent of the line speed; the rest is used for control plane traffic. When RSVP is configured on an interface, the option is to use the entire usable bandwidth or a certain configured amount of bandwidth. The default is for RSVP data flows to use up to 75 percent of the available bandwidth.

The figure shows that when RSVP is configured, a priority queue is set up and used by RSVP data flows. In the event of congestion, all other data flows are scheduled and queued using either WFQ or WRED mechanisms. The type of scheduling and queuing depends on whether reservations are guaranteed-rate or controlled-load QoS services.

When RSVP is used with LLQ and class-based weighted fair queuing (CBWFQ), the RSVP data flows are subject to the same rules as all other traffic.

Enabling RSVP on an Interface

This topic describes how to enable RSVP on an interface.

Enabling RSVP

```
router(config-if)#  
ip rsvp bandwidth [interface-kbps] [single-flow-kbps]
```

- Enables RSVP

```
router(config-if)#  
ip rsvp resource-provider none
```

- Disables reserving interface resources

```
router(config-if)#  
ip rsvp data-packet classification none
```

- Disables packet classification

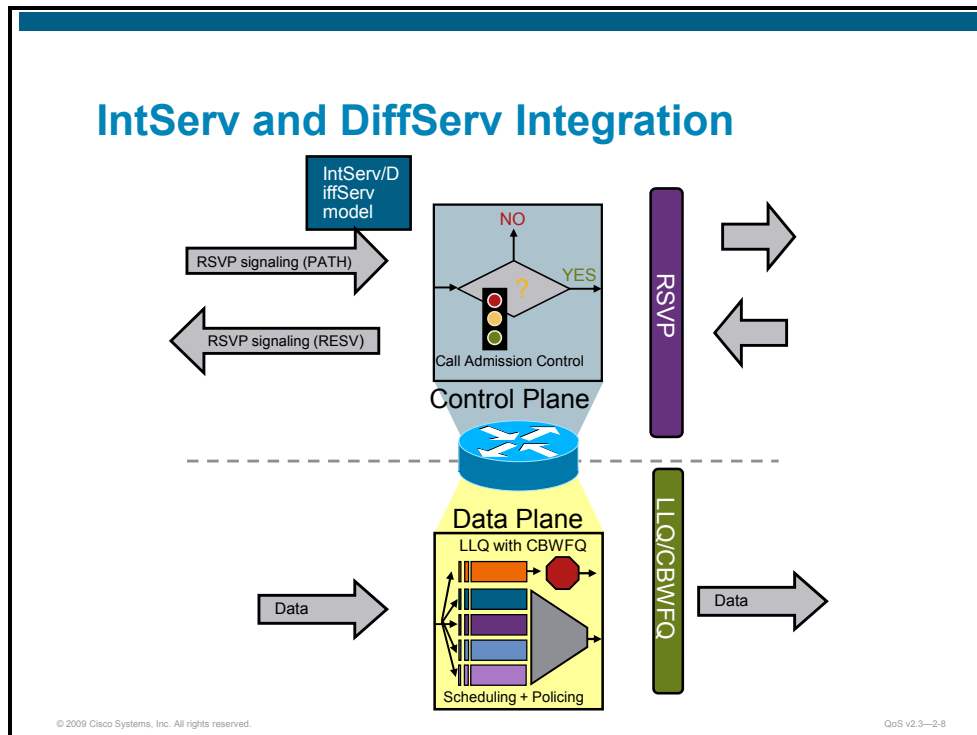
© 2009 Cisco Systems, Inc. All rights reserved. QoS v2.3-2.7

The figure shows the router interface commands used to enable and disable RSVP on an interface. The first bulleted item shows the command for enabling RSVP on an interface, and the arguments under this command are to either accept the default maximum data flow, which is 75 percent of the line speed, or to define a maximum interface bandwidth for RSVP to use and a set maximum for each RSVP data flow.

The second and third bulleted items enable RSVP to follow the same scheduling and queuing mechanisms as configured in LLQ with CBWFQ. These two commands turn off resource reservation protocol, making RSVP only an admission control mechanism.

IntServ and DiffServ Integration

This topic describes how the IntServ module is used in a network.



In the DiffServ model, RSVP involves only the control plane performing admission control, and does not involve the data plane. With IntServ integrated into the DiffServ model, it is possible to add RSVP admission control to a network that is already using a differentiated services approach to QoS. RSVP admits or rejects calls based on a preconfigured bandwidth amount, but the actual scheduling is based on preexisting LLQ criteria, such as the differentiated services code point (DSCP) value of each packet.

Summary

This topic summarizes the key points discussed in this lesson.

Summary

- IntServ provides end-to-end QoS service for applications.
- RSVP is used to provision end-to-end QoS treatment.
- RSVP signaling uses the control plane and is used for Call Admission Control. Reservation data flow uses the data plane in which WFQ and WRED is used.
- The **ip rsvp bandwidth** command is used to enable RSVP on an interface.
- RSVP integrates with DiffServ networks, in which RSVP is used for flow admission control.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-2.9

Understanding the Differentiated Services Model

Overview

Differentiated Services (DiffServ) is a multiple-service model for implementing quality of service (QoS) in the network. With DiffServ, the network tries to deliver a particular kind of service that is based on the QoS specified by each packet. This specification can occur in different ways, such as using the differentiated services code point (DSCP) in IP packets or source and destination addresses. The network uses the QoS specification of each packet to classify, shape, and police traffic and to perform intelligent queuing.

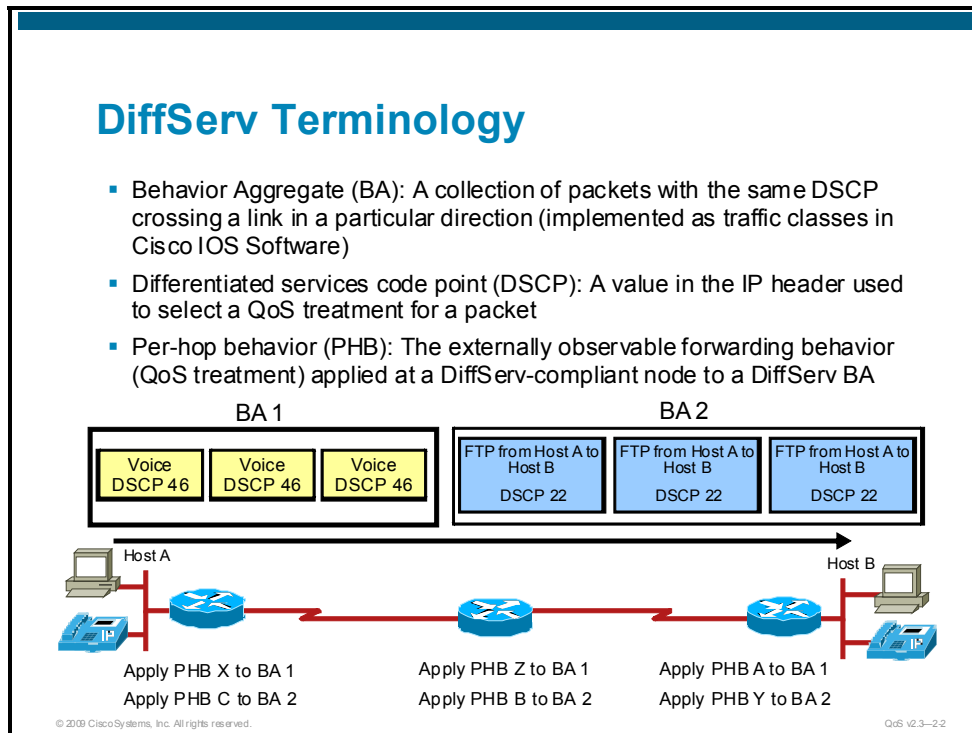
Objectives

Upon completing this lesson, you will be able to describe DiffServ and explain how it can be used to implement QoS in the network. This ability includes being able to meet these objectives:

- Describe the purpose and function of DiffServ
- Describe the basic format and the purpose of the DSCP field in the IP header
- Describe the different per-hop behaviors used in DSCP
- Describe the interoperability between DSCP-based and IP precedence-based devices in a network

Differentiated Services Model

This topic describes the purpose and function of the DiffServ model.



In order to understand the Differentiated Services (DiffServ) module, it is important to understand this DiffServ terminology:

- **Behavior aggregate (BA):** A BA is a collection of packets with the same DSCP value crossing a link in a particular direction. Packets from multiple applications and sources can belong to the same BA. In Cisco IOS Software, classification of packets into BAs can be done by using Modular quality of service command-line interface (MQC) class maps.
- **Differentiated services code point (DSCP):** A value in the IP header that is used to select a QoS treatment for a packet. In the DiffServ model, classification and QoS revolve around the DSCP.
- **Per-hop behavior (PHB):** A PHB is an externally observable forwarding behavior (or QoS treatment) applied at a DiffServ-compliant node to a DiffServ BA. The term PHB refers to the packet scheduling, queueing, policing, or shaping behavior of a node on any given packet belonging to a BA. The DiffServ model itself does not specify how PHBs must be implemented. A variety of techniques may be used to affect the desired traffic conditioning and PHB. In Cisco IOS Software, you can configure PHBs by using MQC policy maps.

DiffServ Model

- In the **DiffServ model**, QoS behaviors are applied to traffic classes on a per-hop basis.
- Complex traffic classification and conditioning are performed at the network edge.
 - Network traffic is categorized into BAs based on the content of some portion of the packet header.
 - Each BA is assigned a **DSCP** value, and the packet header of each packet belonging to the BA is marked with the DSCP value.
- Network devices in the core use the DSCP value to select a **per-hop behavior** for the packet.

IntServ	DiffServ
▪ Flow-based ▪ Each node keeps state information for each flow ▪ Applications request level of service ▪ Scales poorly	▪ Class-based ▪ Per-flow state need not be maintained within the core of the network ▪ Level of service determined by static policies implemented in each node along network path ▪ Scales well

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-2.3

The DiffServ architecture is based on a simple model in which traffic entering a network is classified and possibly conditioned at the boundaries of the network. The traffic class is then identified with a DSCP or bit marking in the IP header. The primary advantage of DiffServ is scalability.

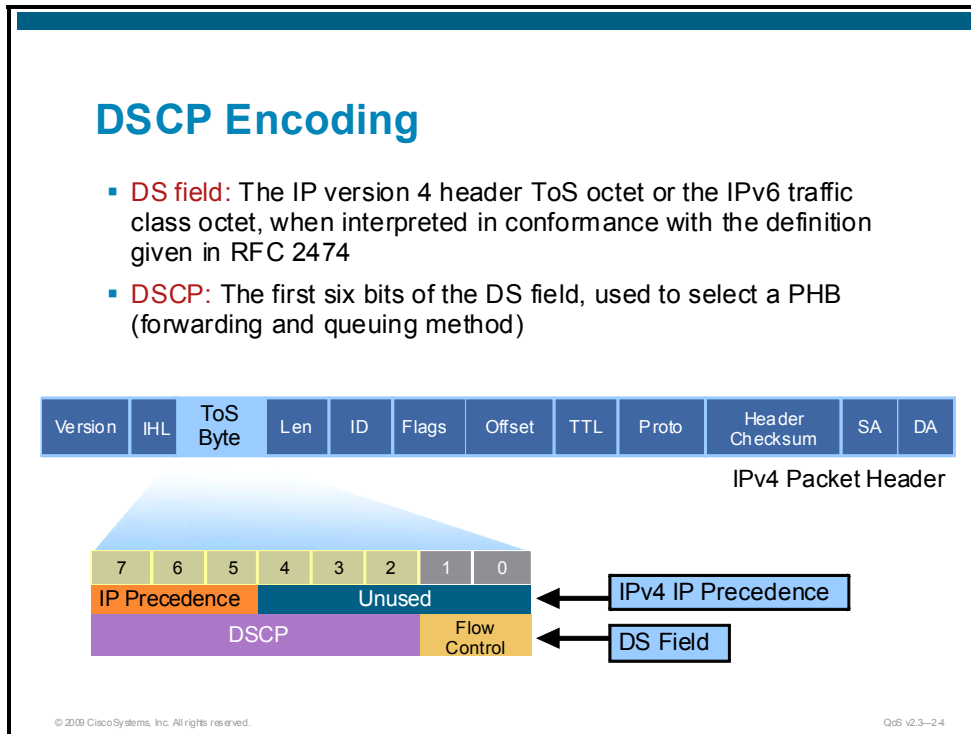
DSCP values are used to mark packets to indicate a desired PHB. Within the core of the network, packets are forwarded according to the PHB that is associated with the DSCP.

One of the primary principles of DiffServ is that you should mark packets as close to the edge of the network as possible. It is often a difficult and time-consuming task to determine the traffic class for a data packet. You should classify the data as few times as possible. By marking the traffic at the network edge, core network devices and other devices along the forwarding path will be able to quickly determine the proper QoS treatment to apply to a given traffic flow.

DSCP supersedes IP precedence, a three-bit field in the type of service (ToS) byte of the IP header originally used to classify and prioritize types of traffic. However, DSCP maintains interoperability with devices that use IP precedence.

DSCP Encoding

This topic describes the basic format of DSCP and explains the purpose of the DSCP field in the IP header.



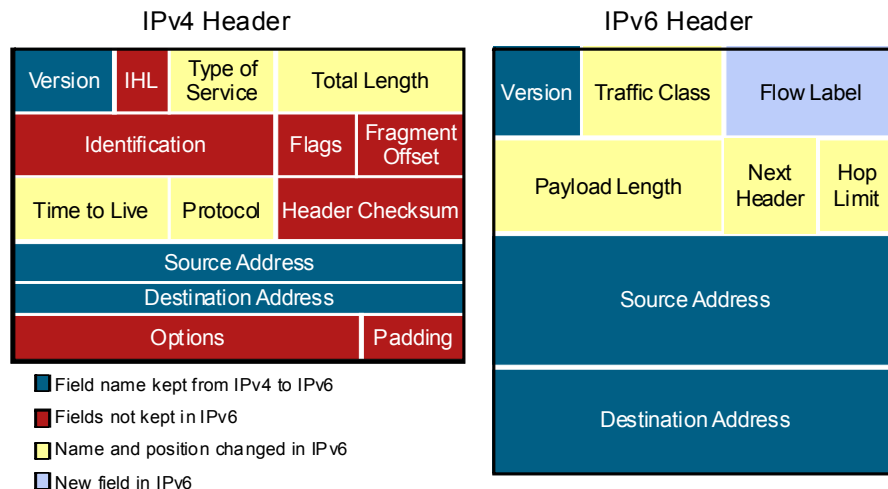
DiffServ uses the Differentiated Services field (DS field) in the IP header to mark packets according to their classification into BAs. The DS field occupies the same eight bits of the IP header that were previously used for the ToS byte.

The following three Internet Engineering Task Force (IETF) standards describe the purpose of the eight bits of the DS field:

- RFC 791 includes specification of the ToS field, where the high-order three bits are used for IP precedence. The other bits are used for delay, throughput, reliability, and cost.
- RFC 1812 modifies the meaning of the ToS field by removing meaning from the five low-order bits (those bits should all be “0”). This gained widespread use and became known as the original IP precedence.
- RFC 2474 replaces the ToS field with the DS field, where the six high-order bits are used for the DSCP. The remaining two bits are used for explicit congestion notification. RFC 3260 (New Terminology and Clarifications for Diffserv) updates RFC 2474 and provides terminology clarifications.

Each DSCP value identifies a BA. Each BA is assigned a PHB. Each PHB is implemented using the appropriate QoS mechanism or set of QoS mechanisms.

IPv6 and QoS



IP version 6 (IPv6) also provides support for QoS marking via a field in the IPv6 header. Similar to the ToS (or DS) field in the IPv4 header, the Traffic Class field (8 bits) is available for use by originating nodes and forwarding routers to identify and distinguish between different classes or priorities of IPv6 packets. The Traffic Class field can be used to set specific precedence or DSCP values, which are used the same way they are used in IPv4.

IPv6 also has a 20-bit field known as the Flow Label field. The flow label enables per-flow processing for differentiation at the IP layer. It can be used for special sender requests and is set by the source node. The flow label must not be modified by an intermediate node. The main benefit of the flow label is that transit routers do not have to open the inner packet to identify the flow, which aids with identification of the flow when using encryption and in other scenarios. The Flow Label field is described in RFC 3697.

Current Cisco IOS Software support for IPv6 QoS includes the following:

- Packet classification
- Queuing (including LLQ)
- Traffic shaping
- Weighted random early detection (WRED)
- Class-based packet marking
- Policing of IPv6 packets

The following QoS features are *not* supported for managing IPv6 traffic:

- Compressed Real-Time Transport Protocol (cRTP)
- Network-Based Application Recognition (NBAR)
- Committed access rate (CAR)
- Priority queuing (PQ)
- Custom queuing (CQ)

Note For more information on implementing QoS for IPv6, refer to the Implementing QoS for IPv6 section of the *Cisco IOS IPv6 Configuration Guide*, Release 12.4, at the following URL:
<http://www.cisco.com/en/US/docs/ios/ipv6/configuration/guide/ip6-qos.html#wp1055373>

Per-Hop Behaviors

This topic describes the different PHBs that are used in DSCP.

Per-Hop Behaviors

DSCP selects PHB throughout the network.

- **Default PHB:** (FIFO, tail drop)
- **EF:** Expedited Forwarding
- **AF:** Assured Forwarding
- **Class-Selector:** (IP Precedence) PHB

101110 DSCP

000 = Default

000 = Class Selector

101 = Expedited Forwarding

001, 010, 011, or 100 = Assured Forwarding

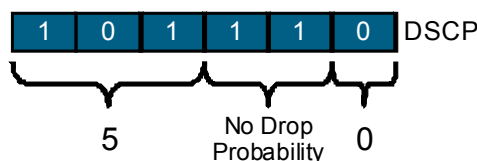
© 2009 Cisco Systems, Inc. All rights reserved. QoS v2.3-2.6

These PHBs are defined by IETF standards:

- **Default PHB:** Used for best-effort service (bits 5 to 7 of DSCP = “000”)
- **Expedited Forwarding (EF) PHB:** Used for low-delay service (bits 5 to 7 of DSCP = “101”)
- **Assured Forwarding (AF) PHB:** Used for guaranteed bandwidth service (bits 5 to 7 of DSCP = “001,” “010,” “011,” or “100”)
- **Class-Selector PHB:** Used for backward compatibility with non-DiffServ-compliant devices (RFC 1812-compliant devices [bits 2 to 4 of DSCP = “000”])

Per-Hop Behaviors (Cont.)

- **EF PHB:**
 - Ensures a minimum departure rate
 - Guarantees bandwidth (The class is guaranteed an amount of bandwidth with prioritized forwarding.)
 - Polices bandwidth (The class is not allowed to exceed the guaranteed amount? excess traffic is dropped.)
- **DSCP value “101110”:** looks like IP precedence 5 to non-DiffServ-compliant devices
 - Bits 5 to 7: “101” = 5 (Same three bits used for IP precedence)
 - Bits 3 to 4: “11” = drop probability high
 - Bit 2: Just “0”



© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-2.7

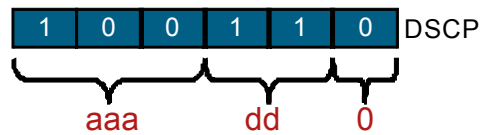
The EF PHB is identified based on the following:

- The EF PHB ensures a minimum departure rate. The EF PHB provides the lowest possible delay to delay-sensitive applications.
- The EF PHB guarantees bandwidth. The EF PHB prevents starvation of the application if there are multiple applications using EF PHB.
- The EF PHB polices bandwidth when congestion occurs. The EF PHB prevents starvation of other applications or classes that are not using this PHB.
- Packets requiring EF should be marked with DSCP binary value “101110” (46 or 0x2E).

Non-DiffServ-compliant devices will regard EF DSCP value “101110” as IP precedence 5 (101). This precedence is the highest user-definable IP precedence and is typically used for delay-sensitive traffic (such as VoIP). Bits 5 to 7 of the EF DSCP value are “101,” which matches IP precedence 5 and allows backward compatibility.

Per-Hop Behaviors (Cont.)

- **AF PHB:**
 - Guarantees bandwidth
 - Allows access to extra bandwidth, if available
- Four standard classes (af1, af2, af3, and af4)
- DSCP value range: “**aaadd0**”
 - Where “aaa” is a binary value of the class
 - Where “dd” is drop probability



© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-2.8

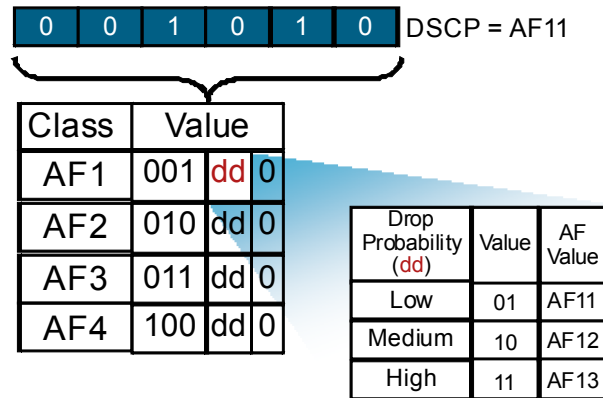
The AF PHB is identified based on the following:

- The AF PHB guarantees a certain amount of bandwidth to an AF class.
- The AF PHB allows access to extra bandwidth, if available.
- Packets requiring AF PHB should be marked with DSCP value “aaadd0,” where “aaa” is the number of the class and “dd” is the drop probability.

There are four standard defined AF classes. Each class should be treated independently and should have allocated bandwidth that is based on the QoS policy.

Per-Hop Behaviors (Cont.)

- Each AF class uses three DSCP values.
- Each AF class is independently forwarded with its guaranteed bandwidth.
- Congestion avoidance is used within each class to prevent congestion within the class.



© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-2.9

Each AF class is assigned an IP precedence and has three drop probabilities: low, medium, and high.

AF_{xy}: Assured Forwarding (RFC 2597), where *x* corresponds to the IP precedence value (only 1–4 are used for AF classes), and *y* corresponds to the Drop Preference value (either 1, 2, or 3).

Per-Hop Behaviors (Cont.)

- A DiffServ node **must** allocate a configurable, minimum amount of forwarding resources (buffer space and bandwidth) per AF class.
- Excess resources may be allocated between non-idle classes. The manner must be specified.
- Reordering of IP packets of the same flow is not allowed if they belong to the same AF class.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3--2-10

An AF implementation must attempt to minimize long-term congestion within each class, while allowing short-term congestion resulting from bursts. This requires an active queue management algorithm. An example of such an algorithm is WRED, which is described in detail in the “Congestion Avoidance” module in this course.

The AF specification does not define the use of a particular algorithm, but it does have certain requirements and properties.

An AF implementation must detect and respond to long-term congestion within each class by dropping packets, while handling short-term congestion (packet bursts) by queuing packets. This implies the presence of a smoothing or filtering function that monitors the instantaneous congestion level and computes a smoothed congestion level. The dropping algorithm uses this smoothed congestion level to determine when packets should be discarded.

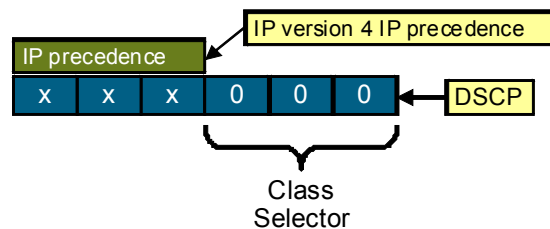
The dropping algorithm must treat all packets within a single class and precedence level identically. Therefore, within a single traffic class, the discard rate of a particular packet flow will be proportional to the percentage of the total amount of traffic.

Backward Compatibility Using the Class Selector

This topic describes the interoperability between DSCP-based and IP precedence-based devices in a network.

Backward Compatibility Using the Class Selector

- Class-Selector “**xx**000” DSCP
- Compatibility with current IP precedence usage (RFC 1812) = maps IP precedence to DSCP
- Differentiates probability of timely forwarding
(**xyz**000) >= (**abc**000) if **xyz** > **abc**
 - If a packet has DSCP = “**01**1000,” it has a greater probability of timely forwarding than a packet with DSCP = “**00**1000.”



© 2003 Cisco Systems, Inc. All rights reserved.

QoS v2.3-2-11

The meaning of the eight bits in the DS field of the IP packet has changed over time to meet the expanding requirements of IP networks.

Originally, the field was referred to as the ToS field and the first three bits of the field (bits 7 to 5) defined a packet IP precedence value. A packet could be assigned one of six priorities based on the value of the IP precedence value (eight total values minus two reserved values). IP precedence 5 (“101”) was the highest priority that could be assigned (RFC 791).

RFC 2474 replaced the ToS field with the DS field, in which a range of eight values (class selector) is used for backward compatibility with IP precedence. There is no compatibility with other bits used by the ToS field.

The Class-Selector PHB was defined to provide backward compatibility for DSCP with ToS-based IP precedence. RFC 1812 simply prioritizes packets according to the precedence value. The PHB is defined as the probability of timely forwarding. Packets with higher IP precedence should be (on average) forwarded in less time than packets with lower IP precedence.

The last three bits of the DSCP (bits 2 to 4), set to “0,” identify a Class-Selector PHB.

Summary

This topic summarizes the key points discussed in this lesson.

Summary

- In the DiffServ model, QoS behaviors are applied to traffic classes on a per-hop basis.
- Network devices in the core use the DSCP value to select a per-hop behavior for the packet.
- A per-hop behavior is an externally observable forwarding behavior applied at a DiffServ-compliant node to a DiffServ behavior aggregate. For example, the EF PHB guarantees and polices bandwidth while ensuring a minimum departure rate.
- The DSCP is backward compatible with IP precedence and Class-Selector Code Point.

Identifying QoS Mechanisms

Overview

Quality of service (QoS) mechanisms are used to implement a coordinated QoS policy in devices throughout the network. The moment an IP packet enters the network, it is classified and usually marked with its class identification. From that point on, the packet is treated by a variety of QoS mechanisms according to the packet classification. Depending upon the mechanisms it encounters, the packet could be expedited, delayed, compressed, fragmented, or even dropped. This lesson describes mechanisms for implementing QoS.

Objectives

Upon completing this lesson, you will be able to correctly match a list of QoS actions to a list of mechanisms for implementing QoS, and identify where in a network the different QoS mechanisms are commonly used. This ability includes being able to meet these objectives:

- Describe the types of key mechanisms used to implement QoS in an IP network
- Describe the purpose of classification and identify where classification is commonly implemented in a network
- Describe the purpose of marking and identify where marking is commonly implemented in a network
- Describe congestion management and identify where congestion management is commonly implemented in a network
- Describe congestion avoidance and identify where congestion avoidance is commonly implemented in a network
- Describe policing and shaping and identify where policing and shaping are commonly implemented in a network
- Describe the functions of compression and identify where compression is commonly implemented in a network
- Describe the functions of LFI and identify where LFI is commonly implemented in a network
- Describe when QoS mechanisms are used for input, output, and both

QoS Mechanisms

This topic describes the types of key mechanisms used to implement QoS in an IP network.

QoS Mechanisms

- **Classification:** Each class-oriented QoS mechanism has to support some type of classification.
- **Marking:** Used to mark packets based on classification, metering, or both.
- **Congestion management:** Each interface must have a queuing mechanism to prioritize transmission of packets.
- **Congestion avoidance:** Used to drop packets early to avoid congestion later in the network.
- **Policing and shaping:** Used to enforce a rate limit based on the metering (excess traffic is either dropped, marked, or delayed).
- **Link efficiency:** Used to improve bandwidth efficiency through compression, link fragmentation, and interleaving.

© 2009 Cisco Systems, Inc. All rights reserved.

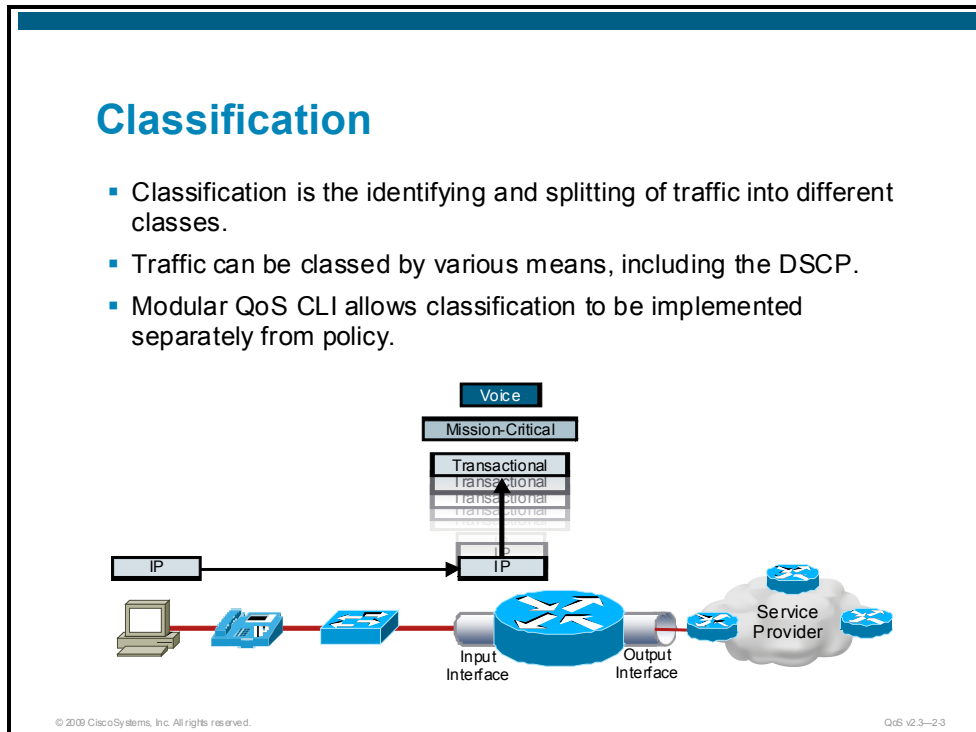
QoS v2.3-2.2

The main categories of tools used to implement QoS in a network are as follows:

- **Classification and marking:** The identifying and splitting of traffic into different classes and the marking of traffic according to behavior and business policies
- **Congestion management:** The prioritization, protection, and isolation of traffic based on markings
- **Congestion avoidance:** Discards specific packets based on markings to avoid network congestion.
- **Policing and shaping:** Traffic conditioning mechanisms that police traffic by dropping misbehaving traffic to maintain network integrity. These mechanisms also shape traffic to control bursts by queuing traffic.
- **Link efficiency:** One type of link efficiency technology is packet header compression, which improves the bandwidth efficiency of a link. Another technology is link fragmentation and interleaving (LFI), which can decrease the “jitter” of voice transmission by reducing voice packet delay.

Classification

This topic describes the purpose of classification and identifies where classification is commonly implemented in a network.



Classification is the identifying and splitting of traffic into different classes. In a QoS-enabled network, all traffic is classified at the input interface of every QoS-aware device. Packet classification can be based on many factors, such as these:

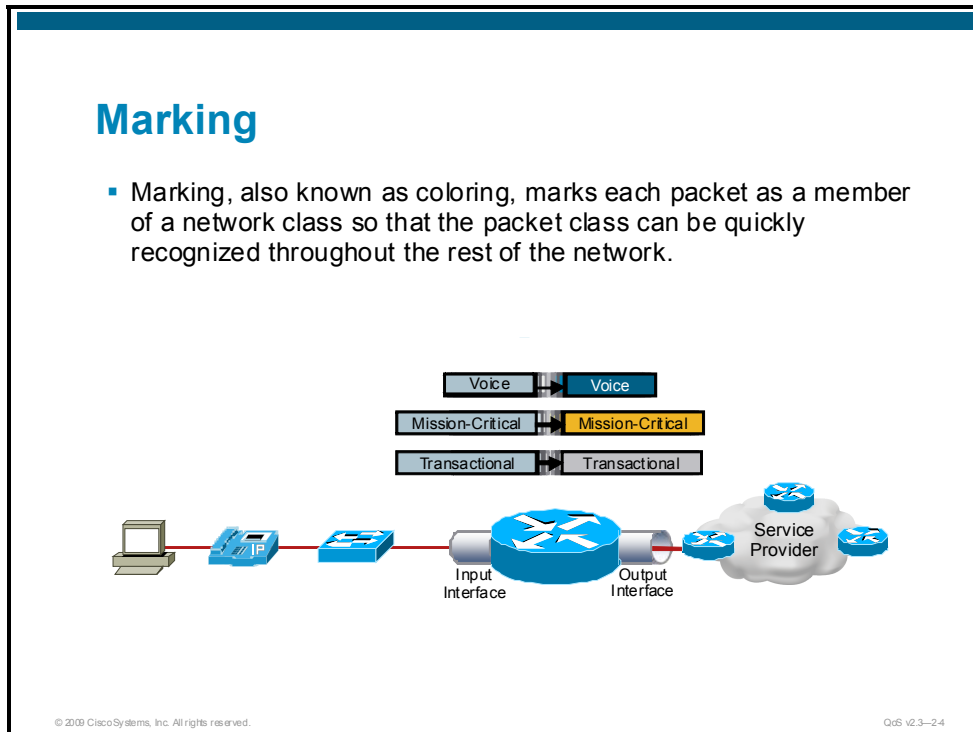
- Differentiated services code point (DSCP)
- IP precedence
- Source address
- Destination address

The concept of trust is key for deploying QoS. When an end device (such as a workstation or an IP phone) marks a packet with class of service (CoS) or DSCP, a switch or router has the option of accepting or not accepting values from the end device. If the switch or router chooses to accept the values, the switch or router trusts the end device. If the switch or router trusts the end device, it does not need to do any reclassification of packets coming from that interface. If the switch or router does not trust the interface, it must perform a reclassification to determine the appropriate QoS value for the packets coming from that interface. Switches and routers are generally set to not trust end devices and must specifically be configured to trust packets coming from an interface. Classification tools include network-based application recognition (NBAR), policy-based routing (PBR), and classification and marking using Modular QoS command-line interface (CLI), or MQC.

Note The tools for classification are covered in detail in the “Classification and Marking” module in this course.

Marking

This topic describes the purpose of marking and identifies where marking is commonly implemented in a network.



Marking, also known as coloring, involves marking each packet as a member of a network class so that devices throughout the rest of the network can quickly recognize the packet class. Marking is performed as close to the network edge as possible and is typically done using MQC.

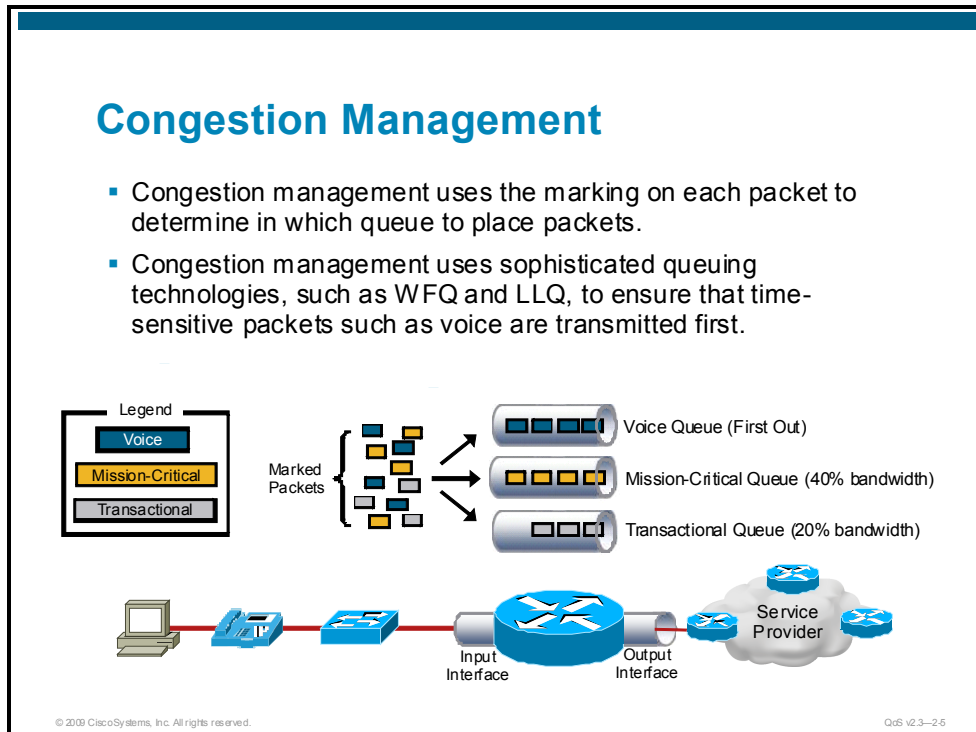
QoS mechanisms set bits in the DSCP or IP precedence fields of each IP packet according to the class that the packet is in. The settings for the DSCP field and their relationship to the IP precedence fields are discussed in the “Understanding the Differentiated Services Model” lesson in this module. Other fields can also be marked to aid in the identification of a packet class.

Other QoS mechanisms use these bits to determine how to treat the packets when they arrive. If the packets are marked as high-priority voice packets, the packets will generally never be dropped by congestion avoidance mechanisms and will be given immediate preference by congestion management queuing mechanisms. On the other hand, if the packets are marked as low-priority file transfer packets, they will be dropped when congestion occurs and will generally be moved to the end of the congestion management queues.

Note The tools for marking are covered in detail in the “Classification and Marking” module in this course.

Congestion Management

This topic describes congestion management and identifies where congestion management is commonly implemented in a network.



Congestion management mechanisms (queuing algorithms) use the marking on each packet to determine in which queue to place packets. Different queues are given different treatment by the queuing algorithm based on the class of packets in the queue. Generally, queues with high-priority packets receive preferential treatment.

Congestion management is implemented on all output interfaces in a QoS-enabled network by using queuing mechanisms to manage the outflow of traffic. Each queuing algorithm was designed to solve a specific network traffic problem and has a particular effect on network performance.

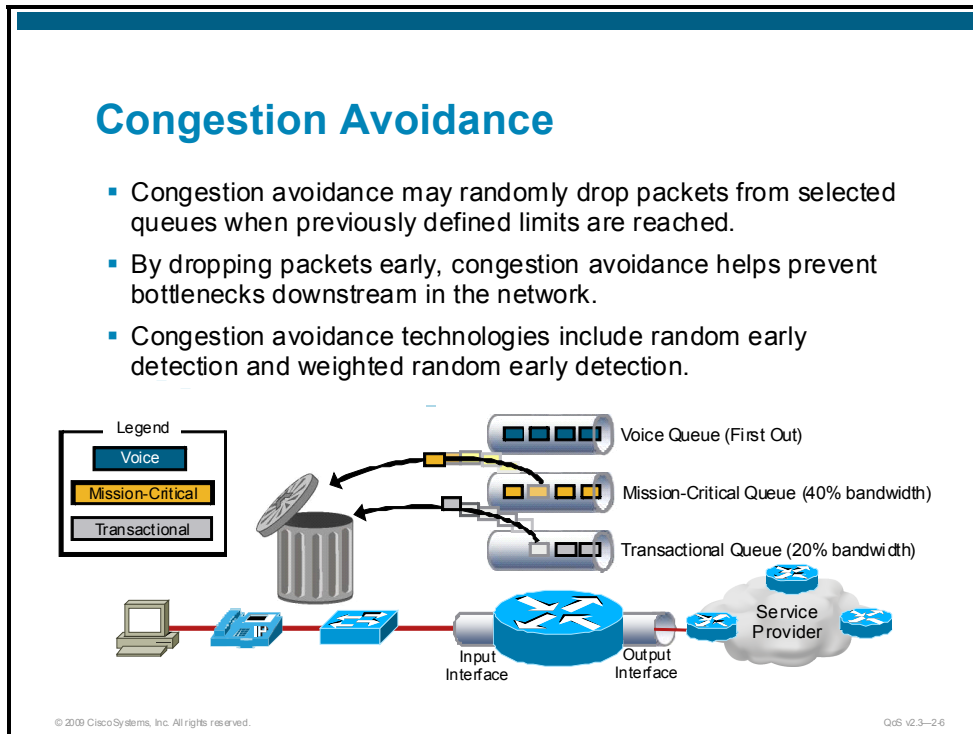
The Cisco IOS Software for congestion management or queuing includes these queuing methods:

- FIFO, priority queuing (PQ), custom queuing (CQ)
- Weighted fair queuing (WFQ)
- Class-based weighted fair queuing (CBWFQ)
- Low-latency queuing (LLQ)

LLQ is currently the preferred queuing method. LLQ is a hybrid (PQ and CBWFQ) queuing method that was developed to specifically meet the requirements of real-time traffic, such as voice.

Congestion Avoidance

This topic describes congestion avoidance and identifies where congestion avoidance is commonly implemented in a network.



Congestion-avoidance mechanisms monitor network traffic loads in an effort to anticipate and avoid congestion at common network bottlenecks. Congestion avoidance is achieved through packet dropping.

Congestion-avoidance mechanisms are typically implemented on output interfaces wherever a high-speed link or set of links feeds into a lower-speed link (such as, a LAN feeding into a slower WAN link.) This ensures that the WAN is not instantly congested by LAN traffic.

Weighted random early detection (WRED) is a Cisco primary congestion-avoidance technique. WRED increases the probability that congestion is avoided by dropping low-priority packets rather than dropping high-priority packets.

WRED is not recommended for voice queues. A network should not be designed to drop voice packets.

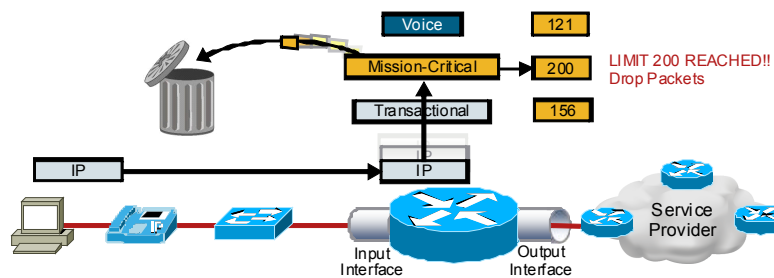
Note The tools for congestion avoidance are covered in detail in the “Congestion Avoidance” module in this course.

Policing and Shaping

This topic describes policing and shaping and identifies where policing and shaping are commonly implemented in a network.

Policing

- Policing drops or marks packets when a predefined limit is reached.



Policing or shaping mechanisms are often used to condition traffic before transmitting traffic to a network or receiving traffic from a network.

Policing is the ability to control bursts and conform traffic to ensure that certain types of traffic get certain types of bandwidth.

Policing drops or marks packets when predefined limits are reached. Policing mechanisms can be set to first drop traffic classes that have lower QoS priority markings.

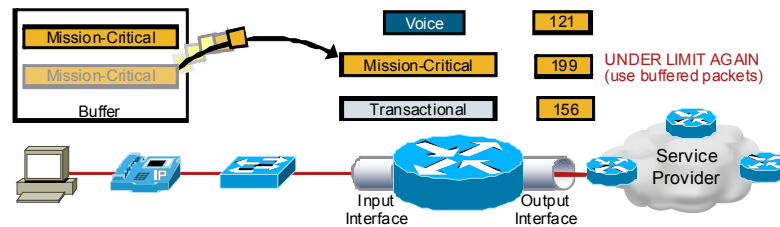
Policing mechanisms can be used at either input or output interfaces. These mechanisms are typically used to control the flow into a network device from a high-speed link by dropping excess low-priority packets. A good example would be the use of policing by a service provider to throttle a high-speed inflow from a customer that was in excess of the service agreement. In a TCP environment, this policing would cause the sender to slow its packet transmission.

Tools include class-based policing and committed access rate (CAR).

Note The tools for policing are covered in detail in the “Traffic Policing and Shaping” module in this course.

Shaping

- Shaping queues packets when a predefined limit is reached.



© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-2.8

Shaping helps smooth out speed mismatches in the network and limits transmission rates.

Shaping mechanisms are used on output interfaces. These mechanisms are typically used to limit the flow from a high-speed link to a lower-speed link to ensure that the lower-speed link does not become overrun with traffic. Shaping could also be used to manage the flow of traffic at a point in the network where multiple flows are aggregated. Service providers use shaping to manage the flow of traffic to and from customers to ensure that the flows conform to service agreements between the customer and provider.

Cisco QoS software solutions include two traffic-shaping tools to manage traffic and congestion on the network: Generic Traffic Shaping (GTS) and Frame Relay traffic shaping (FRTS).

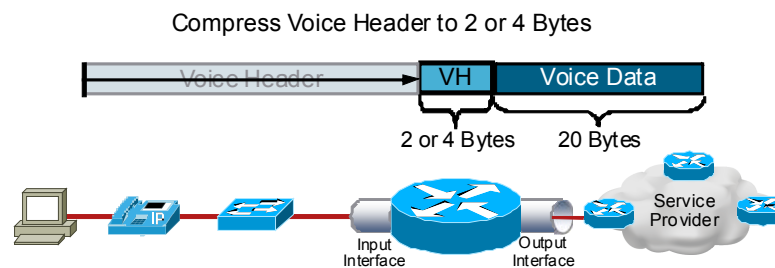
Note The tools for shaping are covered in detail in the “Traffic Policing and Shaping” module in this course.

Link Efficiency Mechanisms

This topic describes the functions of compression and identifies where compression is commonly implemented in the network.

Compression

- Header compression can dramatically reduce the overhead associated with voice transport.



Cisco IOS QoS Software offers link-efficiency mechanisms that work in conjunction with queuing and traffic shaping to manage existing bandwidth more efficiently and predictably. One of these link-efficiency mechanisms is compressed Real-Time Transport Protocol (cRTP).

Real-Time Transport Protocol (RTP) is a host-to-host protocol that is used for carrying converged traffic (including packetized audio and video) over an IP network. RTP provides end-to-end network transport functions intended for applications that transmit real-time requirements, such as audio, video, simulation data multicast, or unicast network services.

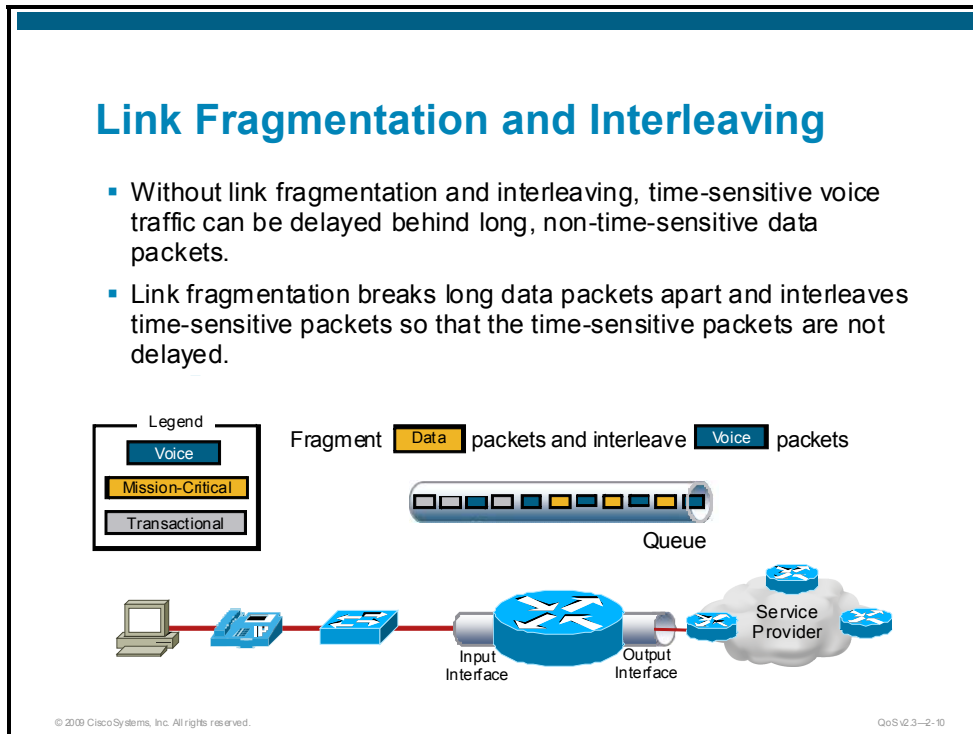
A voice packet carrying a 20-byte voice payload, for example, typically carries a 20-byte IP header, an 8-byte User Datagram Protocol (UDP) header, and a 12-byte RTP header. By using cRTP, as shown in the illustration, the three headers with a combined 40 bytes are compressed down to 2 or 4 bytes, depending on whether the cyclic redundancy check (CRC) is transmitted. This compression can dramatically improve the performance of a link.

Compression would typically be used on WAN links between sites to improve bandwidth efficiency.

Note Compression technology is discussed in the "Link Efficiency Mechanisms" module in this course.

Link Fragmentation and Interleaving

This topic describes the functions of LFI and identifies where LFI is commonly implemented in the network.



Interactive traffic, such as Telnet and VoIP, is susceptible to increased latency and jitter when the network processes large packets, such as LAN-to-LAN FTP Telnet transfers, traversing a WAN link. This susceptibility increases as the traffic is queued on slower links.

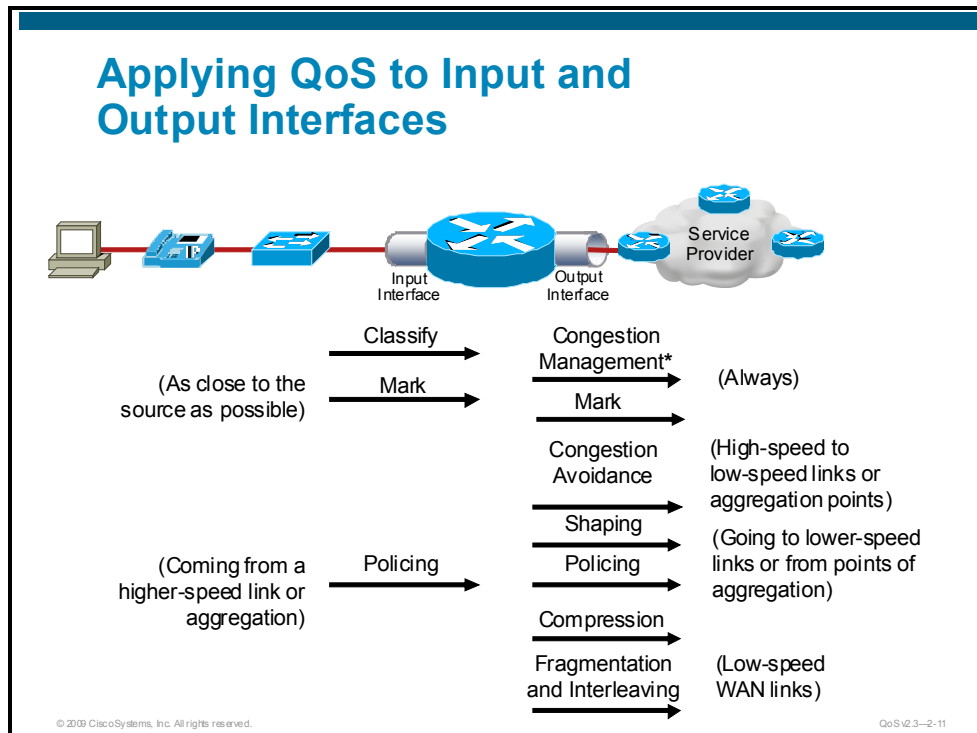
LFI can reduce delay and jitter on slower-speed links by breaking up large datagrams and interleaving low-delay traffic packets with the resulting smaller packets.

LFI would typically be used on WAN links between sites to ensure minimal delay for voice and video traffic.

Note LFI technology is covered in detail in the “Link Efficiency Mechanisms” module in this course.

Applying QoS to Input and Output Interfaces

This topic describes when QoS mechanisms are used for input, output, and both.



In a QoS-enabled network, classification is performed on every input interface. Marking should be performed as close to the network edge as possible—in the originating network device, if possible. Devices farther from the edge of the network, such as routers and switches, can be configured to trust or untrust the markings made by devices on the edge of the network. An IP Phone, for example, will not trust the markings of an attached PC, while a switch will generally be configured to trust the markings of an attached IP Phone.

It only makes sense to use congestion management, congestion avoidance, and traffic-shaping mechanisms on output interfaces, because these mechanisms help maintain smooth operation of links by controlling how much and which type of traffic is allowed on a link. On some router and switch platforms, congestion management mechanisms, such as weighted round robin (WRR) and modified deficit round robin (MDRR), can be applied on the input interface.

Congestion avoidance is typically employed on an output interface wherever there is a chance that a high-speed link or aggregation of links feeds into a slower link (such as a LAN feeding into a WAN).

Policing and shaping are typically employed on output interfaces to control the flow of traffic from a high-speed link to lower-speed links. Policing is also employed on input interfaces to control the flow into a network device from a high-speed link by dropping excess low-priority packets.

Both compression and LFI are typically used on slower-speed WAN links between sites to improve bandwidth efficiency.

Summary

This topic summarizes the key points discussed in this lesson.

Summary

- Different mechanisms can be used to implement QoS in a network: classification, marking, congestion management, congestion avoidance, policing, shaping, and link efficiency.
- The first step is always to identify classes of traffic at the input interface of every QoS-aware device so that the appropriate QoS treatment can be applied to the different traffic types.
- Marking involves marking each packet as a member of a class so that devices throughout the rest of the network can quickly recognize the packet class. Marking is performed as close to the network edge as possible.
- Congestion management is based on placing marked packets into different output interface queues based on their class; these queues are given different treatment based on their associated queuing algorithm.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3—2-12

Summary (Cont.)

- Congestion avoidance mechanisms help prevent link congestion by dropping lower priority traffic at the output interface before it becomes a problem.
- Traffic conditioners such as policers and shapers are used to limit the maximum rate of traffic sent or received on an interface.
- Compression can be used on WAN links between sites to improve bandwidth efficiency.
- Link fragmentation and interleaving can also be used on WAN links to improve bandwidth efficiency.
- QoS mechanisms may be applied at the input interface, output interface, or both, depending on the particular QoS mechanism.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3—2-13

Module Summary

This topic summarizes the key points discussed in this module.

Module Summary

- The three basic models for QoS are best effort, IntServ, and DiffServ. With best effort, no special QoS mechanisms are applied. With IntServ, applications signal to the network that they require QoS. With DiffServ, the network devices recognize the traffic classes of the packets and provide different QoS levels.
- The differentiated services model is the primary model used to implement QoS in IP networks because it is highly scalable and offers the capability to define many different levels of service.
- The differentiated services model uses a 6-bit DSCP to mark packets so that they will be treated with different levels of service as they traverse an IP network.
- IP networks use a variety of mechanisms to implement QoS, including classification, marking, congestion management, congestion avoidance, metering, traffic policing, traffic shaping, and link efficiency.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-2.1

There are three basic networking models: Best Effort, IntServ, and DiffServ. DiffServ is the more scalable model to implement the required QoS for converged networks. With DiffServ, traffic is classified into different traffic classes, and then marked. The network QoS policy then enforces differentiated services based on the markings. To implement the QoS policy, different QoS mechanisms are required.

References

For additional information, refer to these resources:

- To learn more about the IntServ model, refer to *Integrated Services in the Internet Architecture: an Overview* at this URL: <http://www.ietf.org/rfc/rfc1633.txt>.
- To learn more about RSVP, refer to RFC 2210, *The Use of RSVP with IETF Integrated Services* at this URL: <http://www.ietf.org/rfc/rfc2210.txt>.
- To learn more about the DiffServ model, refer to RFC 2475, *An Architecture for Differentiated Services* at this URL: <http://www.ietf.org/rfc/rfc2475.txt>.
- To learn more about the DiffServ field, refer to *Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers* at this URL: <http://www.ietf.org/rfc/rfc2474.txt>.
- To learn more about Assured Forwarding, refer to *Assured Forwarding PHB Group* at this URL: <http://www.ietf.org/rfc/rfc2597.txt>.
- To learn more about Expedited Forwarding, refer to *An Expedited Forwarding PHB (Per-Hop Behavior)* at this URL: <http://www.ietf.org/rfc/rfc3246.txt>.

- To learn more about the Class-Selector PHB, refer to *Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers* at this URL:
<http://www.ietf.org/rfc/rfc2474.txt>.
- To see more information on Cisco QoS mechanisms, select *QoS (Quality of Service)* from the list at <http://www.cisco.com/cisco/web/psa/default.html?mode=tech>.

Module Self-Check

Use the questions here to review what you learned in this module. The correct answers and solutions are found in the Module Self-Check Answer Key.

- Q1) Which of these models for implementing QoS is the least scalable? (Source: Identifying Models for Implementing QoS)
- A) best effort
 - B) Integrated Services
 - C) Differentiated Services
 - D) Quantitative Services
- Q2) Which three QoS mechanisms work together to provide a set of complete integrated services on a network? (Choose three.) (Source: Identifying Models for Implementing QoS)
- A) weighted random early detection
 - B) weighted fair queuing
 - C) Generic Traffic Shaping
 - D) Resource Reservation Protocol
- Q3) What is the most important advantage of DiffServ over other QoS models? (Source: Understanding the Differentiated Services Model)
- A) high scalability
 - B) many service levels
 - C) guaranteed service
 - D) deterministic delays
 - E) advanced queuing mechanisms
- Q4) Services are provided to which entities in the DiffServ model? (Source: Understanding the Differentiated Services Model)
- A) frames
 - B) packets
 - C) applications
 - D) classes of traffic
- Q5) How many bits is the DSCP field of the IP header? (Source: Understanding the Differentiated Services Model)
- A) 3
 - B) 4
 - C) 6
 - D) 8
- Q6) Which PHB would be indicated if the DSCP was equal to 46 (101110)? (Source: Understanding the Differentiated Services Model)
- A) default PHB
 - B) selector PHB
 - C) AF PHB
 - D) EF PHB

- Q7) Which AF class and what drop probability would be indicated if the DSCP was equal to “100100?” (Source: Understanding the Differentiated Services Model)
- A) AF1 and medium
 - B) AF4 and medium
 - C) AF1 and high
 - D) AF4 and high
- Q8) Match each of the following QoS mechanisms to its function. (Source: Identifying QoS Mechanisms)
- A) congestion avoidance
 - B) congestion management
 - C) classification
 - D) traffic policing
 - E) traffic shaping
 - F) packet header compression
- _____ 1. Drops misbehaving traffic to maintain network integrity
- _____ 2. Improves the bandwidth efficiency of a link
- _____ 3. Controls traffic by delaying bursts
- _____ 4. Discards specific packets based on markings
- _____ 5. Identifies and splits traffic
- _____ 6. Prioritizes, protects, and isolates traffic based on markings
- Q9) Which of the following QoS mechanisms is used on both input and output interfaces? (Source: Identifying QoS Mechanisms)
- A) classification
 - B) traffic policing
 - C) traffic shaping
 - D) congestion management

Module Self-Check Answer Key

- Q1) B
- Q2) A, B, D
- Q3) A
- Q4) D
- Q5) C
- Q6) D
- Q7) B
- Q8) 1 = D, 2 = F, 3 = E, 4 = A, 5 = C, 6 = B
- Q9) B

Introduction to MQC and Cisco AutoQoS

Overview

Quality of service (QoS) configurations can be complex. In Cisco IOS Software configurations, there are many different QoS mechanisms, many of which have similar features. Because there are also many options available for providing QoS to different traffic types, the effort to deploy QoS end-to-end in a network infrastructure can easily become overwhelming. Fortunately, Cisco Systems has unified QoS configuration by separating the different components of a QoS policy into different configuration modules. These modules comprise the Cisco Modular QoS CLI, or MQC, which allows network administrators and network implementers to more easily deploy QoS. MQC configurations are consistent for different QoS mechanisms and are therefore easier to learn, deploy, and troubleshoot.

There are some instances, however, in which customers do not want to be concerned with the specifics of QoS configuration. These customers would prefer to enable QoS in the LAN or WAN by using one of two commands and allow the Cisco IOS router and switch to automate the required complex QoS configuration. For those customers, Cisco has developed Cisco AutoQoS VoIP and Cisco AutoQoS for the Enterprise.

This module introduces MQC and Cisco AutoQoS as configuration methods for implementing QoS. This module will also serve as the foundation for more advanced MQC configurations that include additional QoS features and techniques.

Module Objectives

Upon completing this module, you will be able to explain the use of MQC and Cisco AutoQoS to implement QoS on the network. This ability includes being able to meet these objectives:

- Explain how to implement a QoS policy using MQC
- Identify capabilities provided by Cisco AutoQoS VoIP and Cisco AutoQoS for the Enterprise
- Configure QoS on a network using Cisco AutoQoS VoIP
- Describe configuration of QoS on a network using Cisco AutoQoS for the Enterprise

Introducing MQC

Overview

The Modular QoS CLI, or MQC, provides a modular approach to the configuration of quality of service (QoS) mechanisms. Network administrators can introduce new QoS mechanisms and reuse available classification options. This lesson describes how to implement QoS policies using MQC.

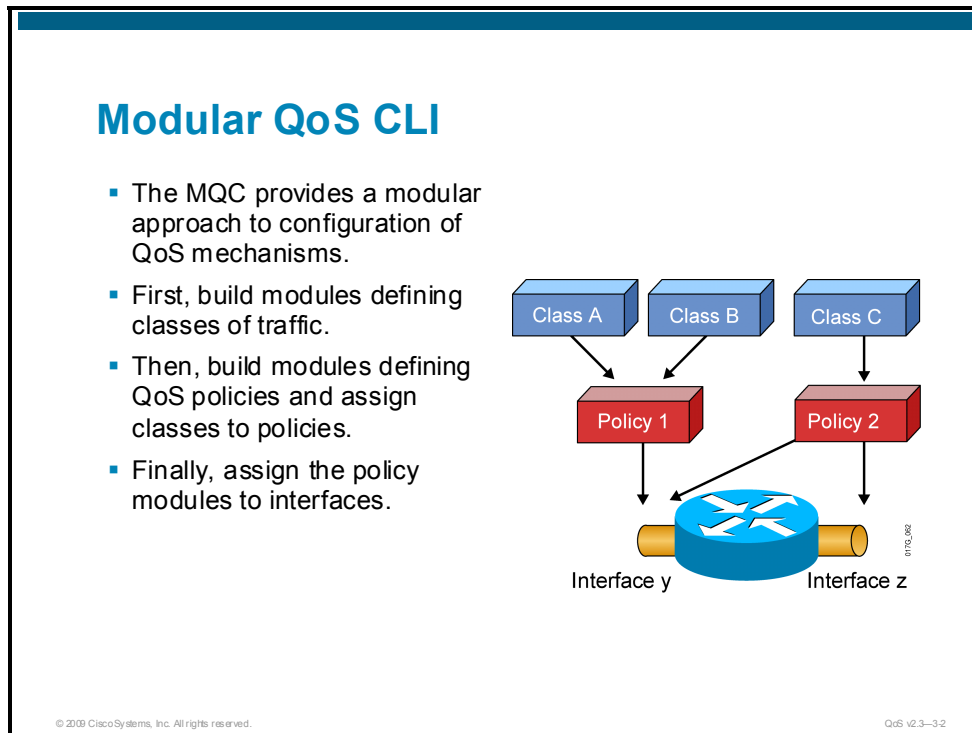
Objectives

Upon completing this lesson, you will be able to explain how to implement a QoS policy using MQC. This ability includes being able to meet these objectives:

- Explain how to implement a given QoS policy using MQC
- Describe the three steps involved in implementing a QoS policy using MQC and the differences between class maps, policy maps, and service policies
- Describe how a class map is used to define a class of traffic
- Describe the Cisco IOS MQC commands required to configure and monitor a class map
- Describe how a policy map is used to assign a QoS policy to a class of traffic
- Describe the Cisco IOS MQC commands required to configure and monitor a policy map
- Explain how a service policy is assigned to an interface
- Describe the MQC commands used to attach a service policy to an interface

Modular QoS CLI

This topic describes how to implement a given QoS policy using MQC.



The MQC was introduced to allow any supported classification to be used with any QoS mechanism.

The separation of classification from the QoS mechanism allows new Cisco IOS Software versions to introduce new QoS mechanisms and reuse all available classification options. On the other hand, old QoS mechanisms can benefit from new classification options.

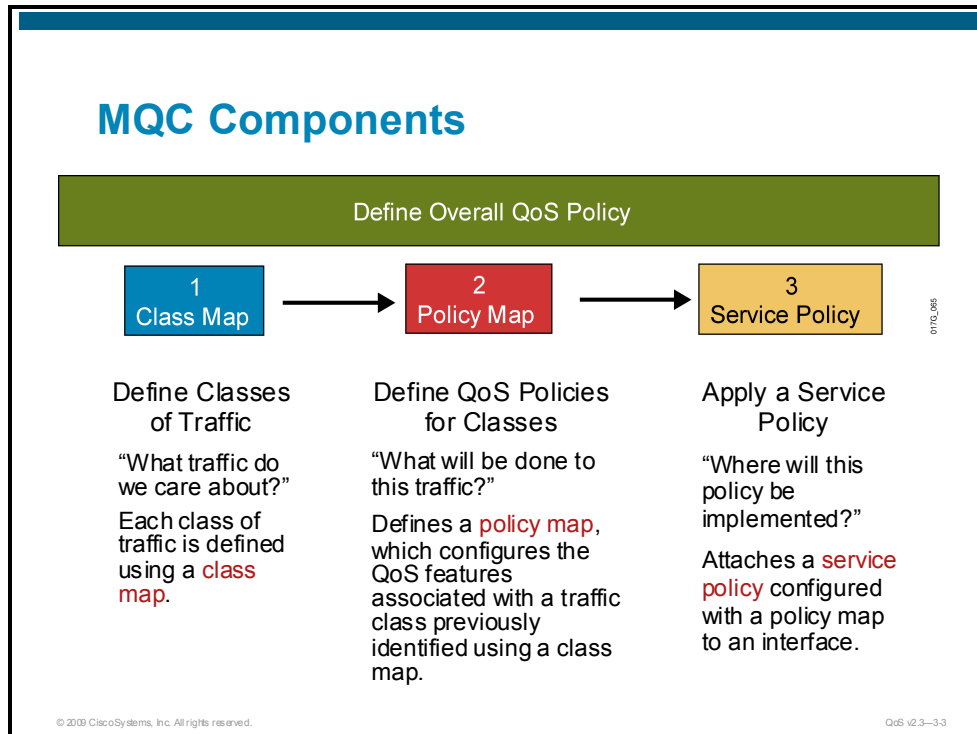
Another important benefit of the MQC is the reusability of configuration. MQC allows the same QoS policy to be applied to multiple interfaces. The MQC, therefore, is a consolidation of all the QoS mechanisms that have so far only been available as standalone mechanisms.

Example: Advantages of Using MQC

Configuring committed access rate (CAR), for example, requires entire configurations to be repeated between interfaces and time-consuming configuration modifications. MQC allows the same QoS policy to be applied to multiple interfaces.

Modular QoS CLI Components

This topic describes the three steps involved in implementing a QoS policy using MQC and differentiates between class maps, policy maps, and service policies.



Implementing QoS by using the MQC consists of three steps:

- Step 1** Configure classification by using the **class-map** command.
- Step 2** Configure traffic policy by associating the traffic class with one or more QoS features using the **policy-map** command.
- Step 3** Attach the traffic policy to inbound or outbound traffic on interfaces, subinterfaces, or virtual circuits by using the **service-policy** command.

Example: Configuring MQC

Consider this example of configuring MQC on a network with voice telephony:

- Step 1** Classify traffic as voice, high priority, low priority, and browser in a class map.
- Step 2** Build a single policy map that defines three different traffic policies (different bandwidth and delay requirements for each traffic class): “NoDelay,” “BestService,” and “Whenever,” and assign the already defined classes of traffic to the policies. Voice is assigned to “NoDelay.” High-priority traffic is assigned to “BestService.” Both low-priority and browser traffic is assigned to “Whenever.”
- Step 3** Assign the policy map to selected router and switch interfaces.

Class Maps

This topic describes how a class map is used to define a class of traffic.

Class Maps

- “What traffic do we care about?”
- Each class is identified using a class map.
- A traffic class contains three major elements:
 - A case-sensitive name
 - A series of **match** commands
 - If more than one **match** command exists in the traffic class, an instruction on how to evaluate these **match** commands
- Class maps can operate in two modes:
 - **Match all**: all conditions have to succeed
 - **Match any**: at least one condition must succeed
- The default mode is **match all**.
- Multiple traffic classes can be configured as a single traffic class (nested).

© 2009 Cisco Systems, Inc. All rights reserved. QoS v2.3-3.4

Class maps are used to create classification templates that are later used in policy maps in which QoS mechanisms are bound to classes.

Routers can be configured with a large number of class maps (currently limited to 256).

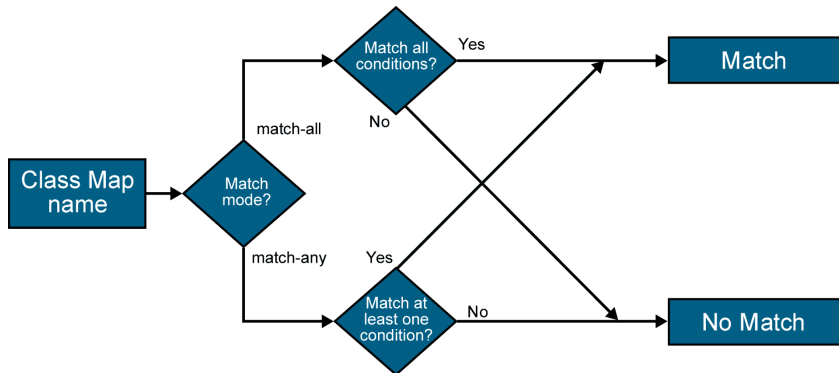
You can create a class map by using the **class-map** global configuration command. Class maps are identified by case-sensitive names. Each class map contains one or more conditions that determine if the packet belongs to the class.

There are two ways of processing conditions when there is more than one condition in a class map:

- **Match all**: This is the default match strategy. All conditions have to be met to bind a packet to the class.
- **Match any**: At least one condition has to be met to bind the packet to the class.

Classification Using Class Maps

- **Match-all** requires all conditions to return a positive answer. If one condition is not met, the class map will return a “no match” result.
- **Match-any** requires at least one condition to return a positive answer. If no condition is met, the class map will return a “no match” result.



The figure illustrates the process of determining if a packet belongs to a class (match) or not (no match).

The process goes through the list of conditions and returns the following:

- A “match” results if one of the conditions is met and the match-any strategy is used.
- A “match” results if all conditions are met and the match-all strategy is used.
- A “no match” results if none of the conditions are met.

Configuring and Monitoring Class Maps

This topic describes the Cisco IOS MQC commands required to configure and monitor a class map.

Configuring Class Maps

```
router(config)#  
class-map [match-all | match-any] class-map-name
```

- Enter the class-map configuration mode.
- Specify the matching strategy.
- Match-all is the default matching strategy.

```
router(config-cmap)#  
match condition
```

- Use at least one condition to match packets.

```
router(config-cmap)#  
description description
```

- You should use descriptions in large and complex configurations.
- The description has no operational meaning.

© 2003 Cisco Systems, Inc. All rights reserved. QoS v2.3-3.6

Use the **class-map** global configuration command to create a class map and enter the class-map configuration mode. A class map is identified by a case-sensitive name; therefore, all subsequent references to the class map must use exactly the same name.

At least one **match** command should be used within the class-map configuration mode (**match none** is the default).

The **description** command is used for documenting a comment about the class map.

Note The syntax shown in the figure is compatible with Cisco IOS Software Release 12.4 and Cisco Catalyst 2960 Switch IOS Software Release 12.2(46)SE. For the command syntax for other software versions or platforms, refer to the appropriate Command Reference on <http://www.cisco.com>.

Example: Class Map Configuration

This example shows a traffic class configured with the **class-map match-all** command:

```
Router(config)# class-map match-all cisco1  
Router(config-cmap)# match protocol ip  
Router(config-cmap)# match qos-group 4  
Router(config-cmap)# match access-group 101
```


If a packet arrives on a router with traffic class called cisco1 configured on the interface, the packet is evaluated to determine if it matches the IP protocol, QoS group 4, *and* access group 101. If all three of these match criteria are met, the packet matches traffic class cisco1.

Configuring Classification Using Special Options

```
router(config-cmap)#
```

```
match not condition
```

- The **not** keyword inverts the condition.

```
router(config-cmap)#
```

```
match class-map class-map-name
```

- One class map can use another class map for classification.
- Nested class maps allow generic template class maps to be used in other class maps.

```
router(config-cmap)#
```

```
match any
```

- The **any** keyword can be used to match all packets.

© 2009 Cisco Systems, Inc. All rights reserved.QoS v2.3-3.7

You can use **match** commands to specify various criteria for classifying packets. Packets are checked to determine whether they match the criteria specified in the **match** commands. If a packet matches the specified criteria, that packet is considered a member of the class and is forwarded according to the QoS specifications set in the traffic policy. Packets that fail to meet any of the matching criteria are classified as members of the default traffic class. The MQC does not necessarily require that you associate a single traffic class to one traffic policy. You can associate multiple types of traffic with a single traffic class by using the **match any** command.

The **match not** command inverts the condition specified. This command specifies a match criterion value that prevents packets from being classified as members of a specified traffic class. All other values of that particular match criterion belong to the class.

The MQC allows multiple traffic classes (nested traffic classes, which are also called nested class maps) to be configured as a single traffic class. This nesting can be achieved with the use of the **match class-map** command. The only method of combining match-any and match-all characteristics within a single traffic class is with the **match class-map** command.

The syntax in the figure is compatible with Cisco IOS Software Release 12.4. The syntax for the **match** command in Cisco Catalyst 2960 Switch IOS Software Release 12.2(46)SE is as follows:

```
match {access-group acl-index-or-name | ip dscp dscp-list | ip  
precedence ip-precedence-list}
```

Parameter	Description
access-group <i>acl-index-or-name</i>	Number or name of an IP standard or extended access control list (ACL) or MAC ACL. For an IP standard ACL, the ACL index range is 1 to 99 and 1300 to 1999. For an IP extended ACL, the ACL index range is 100 to 199 and 2000 to 2699.
ip dscp <i>dscp-list</i>	List of up to eight IP differentiated services code point (DSCP) values to match against incoming packets. Separate each value with a space. The range is 0 to 63. You also can enter a mnemonic name for a commonly-used value.
ip precedence <i>ip-precedence-list</i>	List of up to eight IP-precedence values to match against incoming packets. Separate each value with a space. The range is 0 to 7. You also can enter a mnemonic name for a commonly-used value.

Example: Using the match Command

This example shows a traffic class configured with the **class-map match-any** command:

```
Router(config)# class-map match-any cisco2
Router(config-cmap)# match protocol ip
Router(config-cmap)# match qos-group 4
Router(config-cmap)# match access-group 101
```

In traffic class called cisco2, the match criteria are evaluated consecutively until a successful match criterion is located. The packet is first evaluated to determine whether IP protocol can be used as a match criterion. If IP protocol is not a successful match criterion, then QoS group 4 is evaluated as a match criterion. Each matching criterion is evaluated to see if the packet matches that criterion. When a successful match occurs, the packet is classified as a member of traffic class cisco2. If the packet matches none of the specified criteria, the packet is classified as a member of the “class-default” traffic class.

Example: Nested Traffic Class to Combine match-any and match-all Characteristics in One Traffic Class

The only method for including both match-any and match-all characteristics in a single traffic class is to use the **match class-map** command. To combine match-any and match-all characteristics into a single class, a traffic class created with the **match-any** instruction must use a class configured with the **match-all** instruction as a match criterion (through the **match class-map** command), or vice versa.

This example shows how to combine the characteristics of two traffic classes, one with match-any and one with match-all characteristics, into one traffic class with the **match class-map** command. The result of traffic class class4 requires a packet to match one of these three match criteria to be considered a member of traffic class class4: IP protocol and QoS group 4, destination MAC address 1.1.1, or access group 2.

In this example, only the traffic class called class4 is used with the traffic policy called policy1:

```
Router(config)# class-map match-all class3
Router(config-cmap)# match protocol ip
Router(config-cmap)# match qos-group 4
Router(config-cmap)# exit
Router(config)# class-map match-any class4
Router(config-cmap)# match class-map class3
Router(config-cmap)# match destination-address mac 1.1.1
Router(config-cmap)# match access-group 2
Router(config-cmap)# exit
Router(config)# policy-map policy1
Router(config-pmap)# class class4
Router(config-pmap-c)# police 8100 1500 2504 conform-action
transmit exceed-action set-qos-transmit 4
Router(config-pmap-c)# exit
```

Monitoring Class Maps

router>

```
show class-map [class-name]
```

- Displays all class maps and their matching criteria

```
router>show class-map

Class Map class-3
  Match access-group 103

Class Map class-2
  Match protocol ip

Class Map class-1
  Match input-interface Ethernet1/0
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-3.8

The **show class-map** command lists all class maps with their match statements.

The **show class-map** command with a name of a class map displays the configuration of the selected class map.

The example of **show class-map** in the illustration shows three class maps:

- *class-3* will match any packet to access-group 103.
- *class-2* matches IP packets.
- *class-1* matches any input from interface Ethernet 1/0.

The syntax for the **show class-map** command is as follows:

```
show class-map [class-name]
```

Parameter	Description
<i>class-name</i>	(Optional) Name of the class map. The class map name can be a maximum of 40 alphanumeric characters.

Command syntax varies by device platform and software version, as shown in the following examples. For the command syntax for your platform and software version, refer to the appropriate Command Reference document on <http://www.cisco.com>.

With Cisco IOS Software Release 12.4 on the Cisco 3825 Series Integrated Service Router, you can use the **show class-map** command to display all class maps and their matching criteria; you can use the **show running-config class-map** command to display configured class maps that are part of the currently operating configuration.

The syntax for the **show class-map** and **show run class-map** commands in Cisco IOS Software Release 12.4 on the Cisco 3825 Series Integrated Service Router is as follows:

```
show class-map [class-map-name] [ | { begin | exclude |
include } expression]

show run class-map [class-map-name] [linenum] [ | { begin |
exclude | include } expression]
```

Parameter	Description
<i>class-map-name</i>	(Optional) Name of the class map. The class map name can be a maximum of 40 alphanumeric characters.
linenum	Displays line numbers in output
 begin	Begin with the line that matches <i>expression</i> .
 exclude	Exclude lines that match <i>expression</i> .
 include	Include lines that match <i>expression</i> .
<i>expression</i>	Any regular expression found in show command output

The syntax for the **show class-map** command in Cisco IOS Software Release 12.2(46)SE on the Cisco Catalyst 2960 Switch is as follows:

```
show class-map [class-map-name] [ | {begin | exclude |
include} expression]
```

Parameter	Description
<i>class-map-name</i>	(Optional) Name of the class map. The class map name can be a maximum of 40 alphanumeric characters.
 begin	Begin with the line that matches <i>expression</i> .
 exclude	Exclude lines that match <i>expression</i> .
 include	Include lines that match <i>expression</i> .
<i>expression</i>	Any regular expression found in show command output

Policy Maps

This topic describes how a policy map is used to assign a QoS policy to a class of traffic.

Policy Maps

- “What will be done to this traffic?”
- Defines a traffic policy, which configures the QoS features associated with a traffic class previously identified using a class map.
- A traffic policy contains three major elements:
 - A case-sensitive name
 - A traffic class
 - The QoS policy associated with that traffic class
- Up to 256 traffic classes can be associated with a single traffic policy.
- Multiple policy maps can be nested to influence the sequence of QoS actions.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-3.9

The **policy-map** command is used to create a traffic policy. The purpose of a traffic policy is to configure the QoS features that should be associated with the traffic that has been classified in a user-specified traffic class or classes. A traffic policy contains three elements: a case-sensitive name, a traffic class (specified with the **class** command), and the QoS policies.

The name of a traffic policy is specified in the **policy-map** CLI (for example, issuing the **policy-map class1** command would create a traffic policy named class1). After issuing the **policy-map** command, you are placed into policy-map configuration mode. You can then enter the name of a traffic class, and enter the QoS features to apply to the traffic that matches this class.

The MQC does not necessarily require that you associate only one traffic class to a single traffic policy. When packets match more than one match criterion, multiple traffic classes can be associated with a single traffic policy.

Note	A packet can match only one traffic class within a traffic policy. If a packet matches more than one traffic class in the traffic policy, the first traffic class defined in the policy will be used.

Configuring and Monitoring Policy Maps

This topic describes the Cisco IOS MQC commands required to configure and monitor a policy map.

Configuring Policy Maps

```
router(config)#  
policy-map policy-map-name
```

- Enter policy-map configuration mode.
- Policy maps are identified by a case-sensitive name.

```
router(config-pmap)#  
class {class-name | class-default}
```

- Enter the per-class policy configuration mode by using the name of a previously configured class map.
- Use the name “class-default” to configure the policy for the default class.

```
router(config-pmap)#  
class class-map-name condition
```

- Optionally, you can define a new class map by entering the condition after the name of the new class map.
- Uses the match-any strategy.

© 2008 Cisco Systems, Inc. All rights reserved. QoS v2.2-3.10

You can configure service policies by using the **policy-map** command. Use up to 256 classes within one policy map by using the **class** command with the name of a preconfigured class map.

You can also use a nonexistent class within the policy-map configuration mode if the match condition is specified after the name of the class. The running configuration will reflect such a configuration by using the match-any strategy and inserting a full class map configuration.

The table shows starting and resulting configuration modes for the **class-map**, **policy-map**, and **class** commands:

Configuration Modes

Starting configuration mode	Command	Configuration mode
Router(config)#	class-map	Router(config-cmap)#
Router(config)#	policy-map	Router(config-pmap)#
Router(config-pmap)#	class	Router(config-pmap-c)#

All traffic that is not classified by any of the class maps that are used within the policy map is part of the default class “class-default.” This class has no QoS guarantees, by default. The default class, when used on output, can use one FIFO queue or flow-based weighted fair queuing (WFQ). The default class is part of every policy map, even if not configured.

The syntax shown in the figure for the **policy-map** command is the same as the syntax for the **policy-map** command in both Cisco IOS Software Release 12.2(46)SE (on the Cisco Catalyst 2960 Switch) and Cisco IOS Software Release 12.4 (on the Cisco 3825 Series Integrated Services Router); however, command syntax may vary by device platform and software version.

The syntax shown in the figure for the **class** command is the same as the syntax for the **class** command in both Cisco IOS Software Release 12.2(46)SE (on the Cisco Catalyst 2960 Switch) and Cisco IOS Software Release 12.4 (on the Cisco 3825 Series Integrated Services Router), except that they do not allow you to create a new traffic class within policy-map configuration mode. You must use the **class-map** command in global configuration mode to create classes.

Command syntax varies by device platform and software version. For the command syntax for your platform and software version, refer to the appropriate Command Reference document on <http://www.cisco.com>.

Configuring Policy Maps (Cont.)

```
router(config-pmap) #
```

```
description description
```

- You should use descriptions in large and complex configurations.
- The description has no operational meaning.

```
router(config-pmap-c) #
```

```
<PHB mechanism>
```

- Per-class service policies are configured within the per-class policy-map configuration mode.
- MQC supports the following QoS mechanisms:
 - Class-based weighted fair queuing (CBWFQ)
 - Low-latency queuing
 - Class-based policing
 - Class-based shaping
 - Class-based marking

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-3-11

Policy maps, like class maps, should use descriptions in large QoS implementations where a large number of different policy maps are used.

Renaming a policy map would normally require the renaming of all the references to the policy map. Using the **rename** command simplifies the renaming process by automatically renaming all references.

Example: Policy Map Example

The example shows the configuration of a policy map using three classes. The first two classes were separately configured using the **class-map** command. The third class was configured by specifying the match condition after the name of the class.

```
class-map match-all Test1
  match protocol http
  match access-group 100
class-map match-any Test2
  match protocol http
  match access-group 101
!
policy-map Test
  class Test1
    bandwidth 100
  class Test2
    bandwidth 200
  class Test3 access-group 100
    bandwidth 300
!
access-list 100 permit tcp any host 10.1.1.1
access-list 101 permit tcp any host 10.1.1.2
```

Class Test1 has two match conditions evaluated in the match-all strategy. Classes Test2 and Test3 use the match-any strategy.

Hierarchical (Nested) Policy Maps

```
router(config-pmap-class) #
```

```
service-policy policy-map-name
```

- Policy maps are normally applied to interfaces.
- Nested policy maps can be **applied directly inside other policy maps** to influence sequence of QoS actions.
- For example: **Shape** all traffic to 2 Mb/s; **queue shaped traffic** to provide priority and bandwidth guarantees.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-3-12

The **service-policy** *policy-map-name* command is used to create hierarchical service policies in policy-map class configuration mode.

The **service-policy** [**input** | **output**] *policy-map-name* command is a different command that is used in interface configuration mode. The purpose of the **service-policy** [**input** | **output**] *policy-map-name* command is to attach service policies to interfaces.

The child policy is the previously defined service policy that is now associated with the new service policy through the use of the **service-policy** command. The new service policy that uses the preexisting service policy is called the parent policy. In the hierarchical policy maps example, the service policy named “*child*” is the child policy and the service policy named “*parent*” is the parent policy.

The **service-policy** *policy-map-name* command has the following restrictions:

- The **set** command is not supported on the child policy.
- The **priority** command can be used in either the parent or the child policy, but not both policies simultaneously.
- The **fair-queue** command cannot be defined in the parent policy.

The syntax and usage of the **service-policy** command shown in the figure are compatible with Cisco IOS Software Release 12.4 on the Cisco 3825 Series Integrated Services Router. Cisco IOS Software Release 12.2(46)SE does not support this usage on the Cisco Catalyst 2960 Switch.

Example: Hierarchical Policy Maps

In the example diagram, a child policy map QueueAll is created, which guarantees bandwidth of 1 Mb/s to web traffic.

Hierarchical (Nested) Policy Maps Example

```
class-map AllTraffic
match any
!
policy-map ShapeAll
class AllTraffic
shape 2000000
service-policy QueueAll
!
interface FastEthernet/0
service-policy output ShapeAll
```

```
class-map HTTP
match protocol http
!
policy-map QueueAll
class HTTP
bandwidth 1000
```

Example policy:

- Shape all traffic on Fast Ethernet to 2 Mb/s.
- Out of the 2 Mb/s, guarantee 1 Mb/s to web traffic.

© 2009 Cisco Systems, Inc. All rights reserved. QoS v2.3-3-13

The QueueAll policy map is then nested within a parent policy map named ShapeAll.

Finally, the parent policy map ShapeAll is applied to the FastEthernet interface.

Traffic out of the FastEthernet interface will first be shaped to 2 Mb/s, and then 1 Mb/s of the 2 Mb/s of shaped web traffic will be guaranteed.

Note Additional information on traffic shaping is covered in the “Traffic Policing and Shaping” module in this course.

Example: Hierarchical Policy-Map Configuration

Follow these steps to apply a hierarchical policy:

- Step 1** Create a child or lower-level policy that configures a queuing mechanism. In the example, low-latency queuing (LLQ) is configured using the **priority** command.

```
policy-map child
  class voice
    priority 512
```

- Step 2** Create a parent or top-level policy that applies class-based shaping. Apply the child policy as a command under the parent policy because the admission control for the child class is based on the shaping rate for the parent class.

```
policy-map parent
  class class-default
    shape average 2000000
    service-policy child
```

- Step 3** Apply the parent policy to the subinterface.

```
interface ethernet0/0.1
  service-policy output parent
```

Monitoring Policy Maps

router>

```
show policy-map [policy-map]
```

- Displays the configuration of all classes for a specified service policy map or all classes for all existing policy maps

```
router>show policy-map
Policy Map Test
  Class Test1
    Weighted Fair Queueing
      Bandwidth 100 (kbps) Max Threshold 64 (packets)
  Class Test2
    Weighted Fair Queueing
      Bandwidth 200 (kbps) Max Threshold 64 (packets)
  Class Test3
    Weighted Fair Queueing
      Bandwidth 300 (kbps) Max Threshold 64 (packets)
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS V2.3--3-14

The **show policy-map** command can be used to verify the configuration of a policy map.

With Cisco IOS Software Release 12.4 on the Cisco 3825 Series Integrated Service Router, you can use the **show policy-map** command to display all policy maps and their configurations; you can use the **show running-config policy-map** command to display configured policy maps that are part of the currently operating configuration.

The syntax for the **show policy-map** and **show running-config policy-map** commands in Cisco IOS Software Release 12.4 on the Cisco 3825 Series Integrated Service Router is as follows:

```
show policy-map [policy-map]
```

```
show running-config policy-map [policy-map] [linenum] [|  
{begin | exclude | include}expression]
```

Parameter	Description
<i>policy-map</i>	(Optional) Name of the service policy map whose complete configuration is to be displayed. The name can be a maximum of 40 characters.
linenum	(Optional) Displays line numbers in output
 begin	(Optional) Begin with the line that matches <i>expression</i>
 exclude	(Optional) Exclude lines that match <i>expression</i>
 include	(Optional) Include lines that match <i>expression</i>
<i>expression</i>	(Optional) Any regular expression found in show command output

The syntax for the **show policy-map** command in Cisco IOS Software Release 12.2(46)SE on the Cisco Catalyst 2960 Switch is as follows:

```
show policy-map [policy-map-name [class class-map-name]] [ |  
{begin | exclude | include} expression]
```

Parameter	Description
<i>policy-map-name</i>	(Optional) Name of the service policy map whose complete configuration is to be displayed. The name can be a maximum of 40 characters.
class <i>class-map-name</i>	(Optional) Display QoS policy actions for an individual class
 begin	(Optional) Begin with the line that matches <i>expression</i>
 exclude	(Optional) Exclude lines that match <i>expression</i>
 include	(Optional) Include lines that match <i>expression</i>
<i>expression</i>	(Optional) Any regular expression found in show command output

Monitoring Policy Maps (Cont.)

router>

```
show policy-map interface interface-name [input | output]
```

```
router> show policy-map interface FastEthernet0/0 output
FastEthernet0/0

Service-policy output: Test

Class-map: Test1 (match-any)
  0 packets, 0 bytes
  5 minute offered rate 0 bps, drop rate 0 bps
  Match: access-group 101
  Match: access-group 102
  Match: protocol http
  Weighted Fair Queueing
  Output Queue: Conversation 265
  Bandwidth 100 (Kbps) Max Threshold 64 (packets)
  (pkts matched/bytes matched) 0/0
  (depth/total drops/no-buffer drops) 0/0/0
...
Class-map: class-default (match-any)
  25 packets, 19310 bytes
  5 minute offered rate 1000 bps, drop rate 0 bps
  Match: any
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3--3-15

The **show policy-map** command also displays live information if the *interface* keyword is used. The sample output shows the parameters and statistics of the policy map that is attached to outbound traffic on interface FastEthernet0/0.

The syntax for the **show policy-map interface** command in Cisco IOS Software Release 12.4 on the Cisco 3825 Series Integrated Service Router is as follows:

```
show policy-map interface [interface-type interface-number |
null 0 ] [input | output] [ | {begin | exclude | include}
expression]
```

Parameter	Description
<i>interface-type</i>	(Optional) Interface type; valid values are Async, BVI, CDMA-Ix, CTunnel, Dialer, GigabitEthernet, Group-Async, = Lex, Loopback, MFR, Multilink, Port-channel, Tunnel, Vif , Virtual-PPP, Virtual-Template, Virtual-TokenRing, and XTagATM.
<i>interface-number</i>	(Optional) Module and port number
null 0	(Optional) Specifies the null interface; the only valid value is 0
input	(Optional) Indicates that the statistics for the attached input policy will be displayed
output	(Optional) Indicates that the statistics for the attached output policy will be displayed
 begin	(Optional) Begin with the line that matches <i>expression</i>
 exclude	(Optional) Exclude lines that match <i>expression</i>
 include	(Optional) Include lines that match <i>expression</i>
<i>expression</i>	(Optional) Any regular expression found in show command output

Command syntax varies by device platform and software version. For the command syntax for your platform and software version, refer to the appropriate Command Reference document on <http://www.cisco.com>.

Service Policy

This topic describes how a service policy is assigned to an interface.

Service Policy

- “Where will this policy be implemented?”
- Attaches a traffic policy configured with a policy map to an interface.
- Service policies can be applied to an interface for inbound or outbound packets.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3--3-18

The last configuration step when configuring QoS mechanisms using the MQC is to attach a policy map to the inbound or outbound packets using the **service-policy** command.

Using the **service-policy** command, you can assign a single policy map to multiple interfaces or assign multiple policy maps to a single interface (a maximum of one in each direction, inbound and outbound). A service policy can be applied for inbound or outbound packets.

Attaching Service Policies to Interfaces

This topic describes the MQC commands used to attach a service policy to an interface.

Attaching Service Policies to Interfaces

```
router(config-if)#  
service-policy {input | output} policy-map-name
```

- Attaches the specified service policy map to the input or output interface

```
class-map HTTP  
  match protocol http  
!  
policy-map PM  
  class HTTP  
    bandwidth 20 00  
  class class-default  
    bandwidth 60 00  
!
```

```
interface Serial0/0  
  service-policy output PM  
!
```

© 2009 Cisco Systems, Inc. All rights reserved. QoS v2.3-3-17

Use the **service-policy** interface configuration command to attach a traffic policy to an interface and to specify the direction in which the policy should be applied (either on packets coming into the interface or packets leaving the interface).

The router immediately verifies the correctness of parameters that are used in the policy map. If there is a mistake in the policy map configuration, the router will display a message explaining what is wrong with the policy map.

The sample configuration shows how a policy map is used to separate HTTP from other traffic. HTTP is guaranteed 2 Mb/s. All other traffic belongs to the default class and is guaranteed 6 Mb/s.

The syntax for the **service-policy** command in Cisco IOS Software Release 12.4 on the Cisco 3825 Series Integrated Service Router is the same as the syntax shown in the figure; however, command options vary by router platform.

With Cisco IOS Software Release 12.2(46)SE on the Cisco Catalyst 2960 Switch, you can apply a policy map only to incoming traffic on a physical port. The syntax for this command in Cisco IOS Software Release 12.2(46)SE on the Cisco Catalyst 2960 Switch is as follows:

```
service-policy input policy-map-name
```

For the command syntax for other software versions or platforms, refer to the appropriate Command Reference document on <http://www.cisco.com>.

Example: Complete MQC Configuration

Traffic Classes Defined

In this example, two traffic classes are created and their match criteria are defined. For the first traffic class, called class1, access control list (ACL) 101 is used as the match criterion. For the second traffic class, called class2, ACL 102 is used as the match criterion. Packets are checked against the contents of these ACLs to determine if they belong to the class.

```
Router(config)# class-map class1
Router(config-cmap)# match access-group 101
Router(config-cmap)# exit

Router(config)# class-map class2
Router(config-cmap)# match access-group 102
Router(config-cmap)# exit
```

Traffic Policy Created

In this example, a traffic policy called policy1 is defined to contain policy specifications for the two classes: class1 and class2. The match criteria for these classes was defined in the traffic classes.

For class1, the policy includes a bandwidth allocation request and a maximum packet count limit for the queue reserved for the class. For class2, the policy specifies only a bandwidth allocation request.

```
Router(config)# policy-map policy1
Router(config-pmap)# class class1
Router(config-pmap-c)# bandwidth 3000
Router(config-pmap-c)# queue-limit 30
Router(config-pmap-c)# exit
Router(config-pmap)# class class2
Router(config-pmap-c)# bandwidth 2000
Router(config-pmap-c)# exit
```

Traffic Policy Attached to an Interface

This example shows how to attach an existing traffic policy, which was created in the preceding section, to an interface. After a traffic policy is defined with the **policy-map** command, the traffic policy can be attached to one or more interfaces to specify the traffic policy for those interfaces by using the **service-policy** command in interface configuration mode. Although the same traffic policy can be assigned to multiple interfaces, each interface can have only one traffic policy attached at the input and a single traffic policy attached at the output.

```
Router(config)# interface e1/1
Router(config-if)# service-policy output policy1
Router(config-if)# exit
Router(config)# interface fa1/0/0
Router(config-if)# service-policy output policy1
Router(config-if)# exit
```

Summary

This topic summarizes the key points discussed in this lesson.

Summary

- MQC is a modular approach to designing and implementing an overall QoS policy.
- Applying an overall QoS policy involves three steps: defining class maps to identify classes of traffic, defining QoS policy maps, and assigning the policy maps to interfaces.
- Each class of traffic is defined in a class-map module.
- The **class-map** global configuration command is used to create a class map and enter the class-map configuration mode. The **show class-map** command lists all class maps with their match statements.
- A policy map module defines a traffic policy, which configures the QoS features associated with a traffic class previously identified using a class map.
- The **policy-map** command is used to create a policy map name and enter policy-map configuration mode.
- A service policy attaches a traffic policy configured with a policy map to an interface.
- The **service-policy** command assigns a single policy map to multiple interfaces or assigns multiple policy maps to a single interface (a maximum of one in each direction, inbound and outbound).

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3—3, 18

Introducing Cisco AutoQoS VoIP

Overview

Cisco AutoQoS represents two innovative technologies, Cisco AutoQoS VoIP and Cisco AutoQoS for the Enterprise, which simplify network administration challenges, reducing quality of service (QoS) complexity, deployment time, and cost in enterprise networks. This lesson focuses on Cisco AutoQoS VoIP. When Cisco AutoQoS is mentioned without VoIP or Enterprise appended to it, it refers to both Cisco AutoQoS VoIP and Cisco AutoQoS for the Enterprise.

Cisco AutoQoS VoIP incorporates value-added intelligence in Cisco IOS Software and Cisco Catalyst software to provision and manage large-scale QoS deployments. It provides QoS provisioning for individual routers and switches, simplifying deployment and reducing human error. Cisco AutoQoS VoIP offers straightforward capabilities to automate VoIP deployments for customers who want to deploy IP telephony but who lack the expertise and staffing to plan and deploy IP QoS and IP services. This lesson explores the capabilities, requirements, and configuration of Cisco AutoQoS VoIP.

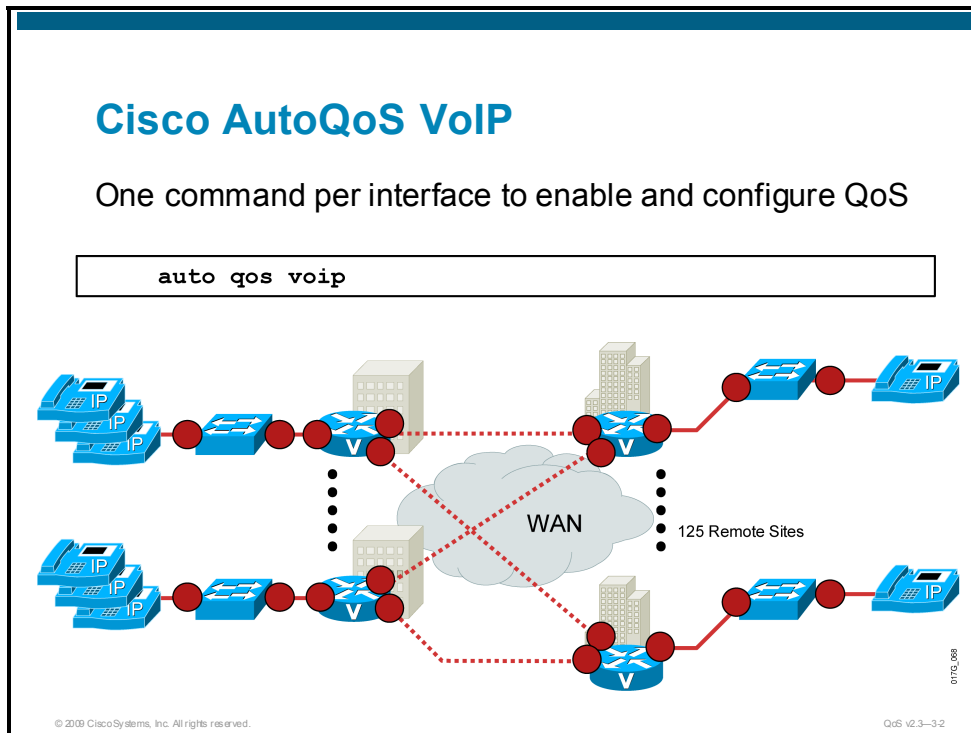
Objectives

Upon completing this lesson, you will be able to identify capabilities provided by Cisco AutoQoS VoIP and configure Cisco AutoQoS VoIP within a network. This ability includes being able to meet these objectives:

- Explain how Cisco AutoQoS is used to implement QoS policy
- Describe the router platforms on which Cisco AutoQoS VoIP can be used
- Describe the switch platforms on which Cisco AutoQoS VoIP can be used
- Describe the prerequisites for using Cisco AutoQoS VoIP and how it is configured on a network using CLI
- Describe how to use Cisco IOS commands to examine and monitor a network configuration after Cisco AutoQoS has been enabled
- Identify several of the QoS technologies that are automatically implemented on the network using Cisco AutoQoS VoIP

Cisco AutoQoS VoIP

This topic describes how you can use Cisco AutoQoS VoIP to implement QoS policy.



Cisco AutoQoS gives you the ability to deploy QoS features for converged IP telephony and data networks fast and efficiently, by simplifying and automating the Modular QoS command-line interface (CLI), or MQC, definition of traffic classes, and by the creation and configuration of traffic policies. Cisco AutoQoS generates traffic classes and policy map CLI templates. When Cisco AutoQoS is configured at the interface or permanent virtual circuit (PVC), the traffic receives the required QoS treatment automatically. You do not need in-depth knowledge of the underlying technologies, service policies, link efficiency mechanisms, and Cisco QoS best practice recommendations for voice requirements to configure Cisco AutoQoS.

Cisco AutoQoS can be beneficial in these scenarios:

- Small- to medium-sized businesses that must deploy IP telephony quickly, but lack the experience and staffing to plan and deploy IP QoS services.
- Large enterprises that need to deploy Cisco telephony solutions on a large scale while reducing the costs, complexity, and time frame for deployment, and ensuring that the appropriate QoS for voice applications is being set in a consistent fashion.
- International enterprises or service providers requiring QoS for VoIP in different regions of the world where little expertise exists and where provisioning QoS remotely and across different time zones is difficult.
- Service providers requiring a template-driven approach for delivering managed services and QoS for voice traffic to many customer premise devices.

Cisco AutoQoS VoIP (Cont.)

Manual QoS

```
interface Multilink1
 ip address 10.1.61.1 255.255.255.0
 ip tcp header-compression iphc-format
 load-interval 30
 service-policy output QoS-Policy
 ppp multilink
 ppp multilink fragment-delay 10
 ppp multilink interleave
 multilink-group 1
 ip rtp header-compression iphc-format
!
interface Serial0
 bandwidth 256
 no ip address
 encapsulation ppp
 no ip mroute-cache
 load-interval 30
 no fair-queue
 ppp multilink
 multilink-group 1
```

AutoQoS

```
interface Serial0
 bandwidth 256
 ip address 10.1.61.1 255.255.255.0
 auto qos voip
```

Cisco AutoQoS automatically creates the QoS-specific features required for supporting the underlying transport mechanism and link speed of an interface or PVC type. For example, Frame Relay traffic shaping (FRTS) is automatically configured and enabled by Cisco AutoQoS for Frame Relay links. Link fragmentation and interleaving (LFI) and compressed Real-Time Transport Protocol (cRTP) are automatically configured via the Cisco AutoQoS template for slow link speeds (less than 768 kbps). Therefore, it is very important that the bandwidth statement be properly set on the interface *prior* to configuring Cisco AutoQoS, because the resulting configuration will vary based on this configurable parameter.

Using Cisco AutoQoS, VoIP traffic is automatically provided with the required QoS template for voice traffic by configuring the **auto qos voip** command on an interface or PVC. Cisco AutoQoS enables the required QoS based on Cisco best-practice methodologies. You can modify the configuration generated by Cisco AutoQoS if necessary.

Cisco AutoQoS VoIP (Cont.)

- Application classification
 - Automatically discovers applications and provides appropriate QoS treatment
- Policy generation
 - Automatically generates initial and ongoing QoS policies
- Configuration
 - Provides high-level business knobs, and multi-device/domain automation for QoS
- Monitoring & reporting
 - Generates intelligent, automatic alerts and summary reports
- Consistency
 - Enables automatic, seamless interoperability among all QoS features and parameters across a network topology – LAN, MAN, and WAN



© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-3.4

Cisco AutoQoS simplifies and shortens the QoS deployment cycle in these five major aspects of successful QoS deployments:

- **Application classification:** Cisco AutoQoS leverages intelligent classification on routers using Cisco network-based application recognition (NBAR) to provide deep and stateful packet inspection. Cisco AutoQoS uses Cisco Discovery Protocol for voice packets to ensure that the device attached to the LAN is really an IP phone.
- **Policy generation:** Cisco AutoQoS evaluates the network environment and generates an initial policy. Cisco AutoQoS automatically determines WAN settings for fragmentation, compression, encapsulation, and Frame Relay-to-ATM Service Interworking (FRF.8), eliminating the need to understand QoS theory and design practices in various scenarios. You can meet additional or special requirements by modifying the initial policy.

The first release of Cisco AutoQoS provides the necessary Cisco AutoQoS VoIP feature to automate QoS settings for VoIP deployments. This feature automatically generates interface configurations, policy maps, class maps, and ACLs. Cisco AutoQoS VoIP will automatically employ Cisco NBAR to classify voice traffic, and mark the traffic with the appropriate differentiated services code point (DSCP) value. You can instruct Cisco AutoQoS VoIP to rely on, or trust, the DSCP markings previously applied to the packets.

- **Configuration:** With one command, Cisco AutoQoS configures the port to prioritize voice traffic without affecting other network traffic, while still offering the flexibility to adjust QoS settings for unique network requirements.

Not only will Cisco AutoQoS automatically detect Cisco IP Phones and enable QoS settings, it will disable the QoS settings when a Cisco IP Phone is relocated or moved to prevent malicious activity.

Cisco AutoQoS-generated router and switch configurations are customizable using the standard Cisco IOS CLI.

- **Monitoring and reporting:** Cisco AutoQoS provides visibility into the classes of service deployed via system logging and Simple Network Management Protocol (SNMP) traps, with notification of abnormal events (that is, VoIP packet drops).
- **Consistency:** When deploying QoS configurations using Cisco AutoQoS, configurations generated are consistent among router and switch platforms. This level of consistency ensures seamless QoS operation and interoperability within the network.

Cisco AutoQoS VoIP Router Platforms

This topic describes the router platforms on which Cisco AutoQoS can be used.

Cisco AutoQoS VoIP: Router Platforms

- Cisco 1760, 1800, 2600, 2800, 3600, 3700, 3800, and 7200 Series routers support Cisco AutoQoS.
- You can meet voice QoS requirements without extensive knowledge of service policies.
- You can reduce the time required to deploy QoS features for converged IP telephony and data networks.
- Cisco AutoQoS lends itself to tuning of all generated parameters & configurations.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-3.6

Support for Cisco AutoQoS includes the Cisco 1760, 1800, 2600 (including XM models), 2800, 3600, 3700, 3800, and 7200 Series routers. Support for additional platforms will become available.

The Cisco AutoQoS VoIP feature is supported only on the following interfaces and PVCs:

- Serial interfaces with PPP or High-Level Data Link Control (HDLC)
- Frame Relay data-link connection identifiers (DLCIs)—point-to-point subinterfaces only
 - Cisco AutoQoS does not support Frame Relay multipoint interfaces
- ATM PVCs
 - Cisco AutoQoS VoIP is supported on low-speed ATM PVCs on point-to-point subinterfaces only (link bandwidth less than 768 kbps)
 - Cisco AutoQoS VoIP is fully supported on high-speed ATM PVCs (link bandwidth greater than 768 kbps)

Cisco AutoQoS VoIP Switch Platforms

This topic describes the switch platforms on which Cisco AutoQoS can be used.

Cisco AutoQoS VoIP: Switch Platforms

- Cisco Catalyst 2950, 2960, 2970, 3550, 3560, 3750, 4500, and 6500 Series switches support Cisco AutoQoS.
- You can meet the voice QoS requirements without extensive knowledge of:
 - Trust boundary
 - CoS-to-DSCP mappings
 - WRR & PQ scheduling parameters
- You can reduce the time required to deploy QoS features for converged IP telephony and data networks.
- Generated parameters and configurations are user-tunable.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-3.6

Support for Cisco AutoQoS includes the Cisco Catalyst 2950, 2960, 2970, 3550, 3560, 3750, 4500, and 6500 Series switches.

The Enhanced Image (EI) software is required on the Cisco Catalyst 29xx Series switches.

Cisco AutoQoS VoIP: Switch Platforms (Cont.)

- Single command at the interface level configures interface and global QoS
 - Support for Cisco IP phone and Cisco IP SoftPhone
 - Trust boundary is disabled when IP phone is moved
 - Buffer allocation and egress queuing dependent on interface type (Gigabit Ethernet/Fast Ethernet)
- Supported on static, dynamic-access, voice VLAN access, and trunk ports
- Cisco Discovery Protocol must be enabled for Cisco AutoQoS to function properly

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-3.7

To configure the QoS settings and the trusted boundary feature on the Cisco IP phone, Cisco Discovery Protocol version 2 or later must be enabled on the port. If the trusted boundary feature is enabled, a syslog warning message displays if Cisco Discovery Protocol is not enabled or if Cisco Discovery Protocol is running version 1.

Cisco Discovery Protocol needs to be enabled only for the **ciscoipphone** QoS configuration; Cisco Discovery Protocol does not affect the other components of the automatic QoS features. When the **ciscoipphone** keyword with the port-specific automatic QoS feature is used, a warning displays if the port does not have Cisco Discovery Protocol enabled.

When executing the port-specific automatic QoS command with the **ciscoipphone** keyword without the trust option, the trust-device feature is enabled. The trust-device feature is dependent on Cisco Discovery Protocol. If Cisco Discovery Protocol is not enabled or not running version 2, a warning message displays as follows:

```
Console> (enable) set port qos 4/1 autoqos voip ciscoipphone
Warning: CDP is disabled or CDP version 1 is in use. Ensure
that CDP version 2 is enabled globally, and also ensure that
CDP is enabled on the port(s) you wish to configure autoqos
on.
Port 4/1 ingress QoS configured for ciscoipphone.
It is recommended to execute the "set qos autoqos" global
command if not executed previously.
Console> (enable)
```

Configuring Cisco AutoQoS VoIP

This topic describes the prerequisites for using Cisco AutoQoS and how to configure Cisco AutoQoS on a network using CLI.

Configuring Cisco AutoQoS VoIP: Prerequisites for Using Cisco AutoQoS VoIP

- Cisco Express Forwarding must be enabled at the interface or ATM PVC.
- This feature cannot be configured if a QoS policy (service policy) is attached to the interface.
- An interface is classified as low-speed if its bandwidth is less than or equal to 768 kb/s. It is classified as high-speed if its bandwidth is greater than 768 kb/s.
 - The correct bandwidth should be configured on all interfaces or subinterfaces using the **bandwidth** command.
 - If the interface or subinterface has a link speed of 768 kb/s or lower, an IP address must be configured using the **ip address** command.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-3.8

Before configuring Cisco AutoQoS, you must meet these prerequisites:

- Cisco Express Forwarding must be enabled at the interface, ATM PVC, or Frame Relay DLCIs (point-to-point subinterfaces only). Cisco AutoQoS uses NBAR to identify various applications and traffic types, and Cisco Express Forwarding is a prerequisite for NBAR.
- You must ensure that no QoS policies (service policies) are attached to the interface. Cisco AutoQoS cannot be configured if a QoS policy (service policy) is attached to the interface.
- Cisco AutoQoS classifies links as either low-speed or high-speed depending upon the link bandwidth. Remember that on a serial interface, if the default bandwidth is not specified, it is 1.544 Mb/s. Therefore, it is important that the correct bandwidth be specified on the interface or subinterface where Cisco AutoQoS is to be enabled.
 - For all interfaces or subinterfaces, be sure to properly configure the bandwidth by using the **bandwidth** command. The amount of bandwidth that is allocated should be based on the link speed of the interface.
 - If the interface or subinterface has a link speed of 768 kb/s or lower, an IP address must be configured on the interface or subinterface using the **ip address** command. By default, Cisco AutoQoS will enable Multilink PPP (MLP) and copy the configured IP address to the multilink bundle interface.

In addition to the Cisco AutoQoS prerequisites, the following are recommendations and requirements when configuring Cisco AutoQoS. Be aware that these may change with Cisco IOS Software releases and should be verified before implementing Cisco AutoQoS in the environment.

- The Cisco AutoQoS VoIP feature is supported only on PVCs and these interfaces:
 - ATM PVCs
 - Serial interfaces with PPP or HDLC
 - Frame Relay DLCIs (point-to-point subinterfaces only) (Cisco AutoQoS does not support Frame Relay multipoint interfaces.)
- Configuration template generated by configuring Cisco AutoQoS on an interface or PVC can be tuned manually, via CLI configuration, if desired.
- Cisco AutoQoS cannot be configured if a QoS service policy is already configured and attached to the interface or PVC.
- MLP is configured automatically for a serial interface with a low-speed link. The serial interface must have an IP address, which is removed and put on the MLP bundle. Cisco AutoQoS VoIP must also be configured on the other side of the link.
- The **no auto qos voip** command removes Cisco AutoQoS. However, if the interface or PVC upon which Cisco AutoQoS generated QoS configuration is deleted without configuring the **no auto qos voip** command, Cisco AutoQoS VoIP will not be completely removed from the configuration.
- Cisco AutoQoS SNMP traps are only delivered when an SNMP server is used in conjunction with Cisco AutoQoS.
- The SNMP community string “AutoQoS” should have “write” permissions.
- If the device is reloaded with the saved configuration after configuring Cisco AutoQoS and saving the configuration to NVRAM, some warning messages may be generated by Remote Monitoring (RMON) threshold commands. These warning messages can be ignored. (To avoid further warning messages, save the configuration to NVRAM again without making any changes to the QoS configuration.)
- By default, Cisco 7200 Series routers and below that support MQC QoS reserve up to 75 percent of the interface bandwidth for user-defined classes. The remaining bandwidth is used for the default class. However, the entire remaining bandwidth is *not* guaranteed to the default class. This bandwidth is shared proportionally between the different flows in the default class and excess traffic from other bandwidth classes. At least one percent of the available bandwidth is reserved and guaranteed for class default traffic by default on Cisco 7500 Series routers. (Up to 99 percent can be allocated to the other classes.)

Configuring Cisco AutoQoS VoIP: Routers

```
router(config-if)# or router(config-fr-dlci)#
```

```
auto qos voip [trust] [fr-atm]
```

- Configures the Cisco AutoQoS VoIP feature
- Untrusted mode by default
- **trust**: Indicates that the DSCP markings of a packet are trusted (relied on) for classification of the voice traffic
- **fr-atm**: For low-speed Frame Relay DLCIs interconnected with ATM PVCs in the same network, the **fr-atm** keyword must be explicitly configured in the **auto qos voip** command to correctly configure the Cisco AutoQoS VoIP feature.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-3.9

To configure the Cisco AutoQoS VoIP feature on an interface, use the **auto qos voip** command in interface configuration mode or Frame Relay DLCI configuration mode. To remove the Cisco AutoQoS VoIP feature from an interface, use the **no** form of the **auto qos voip** command.

The syntax for the **auto qos voip** command is as follows:

```
auto qos voip [trust] [fr-atm]
```

Syntax Description

Parameter	Description
trust	(Optional) Indicates that the DSCP markings of a packet are trusted (relied on) for classification of the voice traffic. If the optional trust keyword is not specified, the voice traffic is classified using NBAR, and the packets are marked with the appropriate DSCP value.
fr-atm	(Optional) Enables the Cisco AutoQoS VoIP feature for the Frame Relay-to-ATM links. This option is available on the Frame Relay DLCIs for Frame Relay-to-ATM Interworking only.

Note This syntax is compatible with Cisco IOS Release 12.4.

The bandwidth of the serial interface is used to determine the link speed. The link speed is one element that is used to determine the configuration generated by the Cisco AutoQoS VoIP feature. The Cisco AutoQoS VoIP feature uses the bandwidth at the time the feature is configured and does not respond to changes made to bandwidth after the feature is configured.

For example, if the **auto qos voip** command is used to configure the Cisco AutoQoS VoIP feature on an interface with 1000 kb/s, the Cisco AutoQoS VoIP feature generates configurations for high-speed interfaces. However, if the bandwidth is later changed to 500 kb/s, the Cisco AutoQoS VoIP feature will not use the lower bandwidth. The Cisco AutoQoS VoIP feature retains the higher bandwidth and continues to use the generated configurations for high-speed interfaces.

To force the Cisco AutoQoS VoIP feature to use the lower bandwidth (and thus generate configurations for the low-speed interfaces), use the **no auto qos voip** command to remove the Cisco AutoQoS VoIP feature and then reconfigure the feature.

Example: Configuring the Cisco AutoQoS VoIP Feature on a High-Speed Serial Interface

In the example, the Cisco AutoQoS VoIP feature is configured on the high-speed serial interface s1/2:

```
Router> enable
Router# configure terminal
Router(config)# interface s1/2
Router(config-if)# bandwidth 1540
Router(config-if)# auto qos voip
Router(config-if)# exit
```

Configuring Cisco AutoQoS VoIP: Cisco Catalyst 6500 Switch

```
Console> (enable)
```

```
set qos autoqos
```

- Global configuration command.
- All the global QoS settings are applied to all ports in the switch.
- Prompt displays the CLI for the port-based automatic QoS commands currently supported.

```
Console> (enable) set qos autoqos
```

```
QoS is enabled
```

```
.....
```

```
All ingress and egress QoS scheduling parameters configured on all  
ports. CoS to DSCP, DSCP to CoS, IP Precedence to DSCP and policed  
dscp maps configured.
```

```
Global QoS configured, port specific autoqos recommended:
```

```
set port qos <mod/port> autoqos trust <cos|dscp>
```

```
set port qos <mod/port> autoqos voip <ciscoipphone|ciscosoftphone>
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS V2.3-3-10

When the global automatic QoS macro is executed, all the global QoS settings are applied to all ports in the switch. After completion, a prompt will display showing the CLI for the port-based automatic QoS commands that are currently supported.

Configuring Cisco AutoQoS VoIP: Cisco Catalyst 6500 Switch (Cont.)

Console> (enable)

```
set port qos mod/port autoqos trust [cos | dscp]
```

- **trust dscp** and **trust cos** are automatic QoS keywords used for ports requiring a “trust all” type of solution.
- **trust dscp** should be used only on ports that connect to other switches or known servers, because the port will be trusting all inbound traffic marking Layer 3 (DSCP).
- **trust cos** should only be used on ports connecting other switches or known servers, because the port trusts all inbound traffic marking in Layer 2 (CoS).
- The trusted boundary feature is disabled and no QoS policing is configured on these types of ports.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3—3-11

The port-specific automatic QoS macro handles all inbound QoS configurations specific to a particular port.

The QoS ingress port-specific settings include port trust, default class of service (CoS), classification, and policing, but does not include scheduling. Input scheduling is programmed through the global automatic QoS macro. Together with the global automatic QoS macro command, all QoS settings are configured properly for a specific QoS traffic type.

Any existing QoS ACLs that are already associated with a port are removed when Cisco AutoQoS modifies ACL mappings on that port. The ACL names and instances are not changed.

Configuring Cisco AutoQoS VoIP: Cisco Catalyst 6500 Switch (Cont.)

Console> (enable)

```
set port qos autoqos voip [ciscosoftphone | ciscoipphone]  
mod/port
```

ciscosoftphone

- The trusted boundary feature must be disabled for Cisco SoftPhone ports.
- QoS settings must be configured to trust the Layer 3 markings of the traffic that enters the port.
- Available on most Catalyst platforms.

ciscoipphone

- The port is set up to trust-cos as well as to enable the trusted boundary feature.
- Combined with the global automatic **qos** command, all settings are configured on the switch to properly handle the signaling and voice bearer and PC data entering and leaving the port.
- Cisco Discovery Protocol must be enabled for the ciscoipphone QoS configuration.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-3-12

The port-specific automatic QoS macro accepts a mod/port combination and must include a Cisco AVVID (Cisco Architecture for Voice, Video, and Integrated Data, now Cisco Unified Communications) system type of keyword. The **ciscoipphone**, **ciscosoftphone**, and **trust** keywords are supported.

With the **ciscoipphone** keyword, the port is set up to trust-cos as well as to enable the trusted boundary feature. Combined with the global automatic QoS command, all settings are configured on the switch to properly handle the signaling, voice bearer, and PC data entering and leaving the port.

In addition to the switch-side QoS settings that are covered by the global automatic QoS command, the IP Phone has a few QoS features that need to be configured for proper labeling to occur. QoS configuration information is sent to the IP Phone through Cisco Discovery Protocol from the switch. The QoS values that need to be configured are the trust settings of the “PC port” on the IP Phone (trusted or untrusted) and the CoS value that is used by the IP Phone to re-mark packets in case the port is untrusted (*ext-cos*).

On the ports that connect to a Cisco SoftPhone, QoS settings must be configured to trust the Layer 3 markings of the traffic that enters the port. Trusting all Layer 3 markings is a security risk, because PC users could send nonpriority traffic with DSCP 46 and gain unauthorized performance benefits. Although not configured by Cisco AutoQoS, policing on all inbound traffic can be used to prevent malicious users from obtaining unauthorized bandwidth from the network. Policing is accomplished by rate-limiting the DSCP 46 (Expedited Forwarding [EF]) inbound traffic to the codec rate used by the Cisco SoftPhone application (worst case G.722). Any traffic that exceeds this rate is marked down to the default traffic rate (DSCP 0, or Best Effort). Signaling traffic (DSCP 24) is also policed and marked down to zero if excess signaling traffic is detected. All other inbound traffic types are reclassified to default traffic (DSCP 0, or best effort).

Note You must disable the trusted boundary feature for Cisco SoftPhone ports.

Example: Using the Port-Specific Cisco AutoQoS Macro

This example shows how to use the **ciscoipphone** keyword:

```
Console> (enable) set port qos 3/1 autoqos help  
Usage: set port qos <mod/port> autoqos trust <cos | dscp>  
set port qos <mod/port> autoqos voip  
<ciscoipphone|ciscosoftphone>  
Console> (enable) set port qos 3/1 autoqos voip ciscoipphone  
Port 3/1 ingress QoS configured for Cisco IP Phone.  
It is recommended to execute the "set qos autoqos" global  
command if not executed previously.  
Console> (enable)
```

This example shows how to use the **ciscosoftphone** keyword:

```
Console> (enable) set port qos 3/1 autoqos voip ciscosoftphone  
Port 3/1 ingress QoS configured for Cisco Softphone.  
It is recommended to execute the "set qos autoqos" global  
command if not executed previously.  
Console> (enable)
```

This example shows how to use the **trust cos** keyword:

```
Console> (enable) set port qos 3/1 autoqos trust cos  
Port 3/1 QoS configured to trust all incoming CoS marking.  
It is recommended to execute the "set qos autoqos" global  
command if not executed previously.  
Console> (enable)
```

This example shows how to use the **trust dscp** keyword:

```
Console> (enable) set port qos 3/1 autoqos trust dscp  
Port 3/1 QoS configured to trust all incoming DSCP marking.  
It is recommended to execute the "set qos autoqos" global  
command if not executed previously.  
Console> (enable)
```

Configuring Cisco AutoQoS VoIP: Catalyst 2950 (EI), 3550 Switches

Switch(config-if)#

```
auto qos voip trust
```

- The uplink interface is connected to a trusted switch or router, and the VoIP classification in the ingress packet is trusted.

Switch(config-if)#

```
auto qos voip cisco-phone
```

- Automatically enables the trusted boundary feature, which uses the Cisco Discovery Protocol to detect the presence or absence of a Cisco IP phone.
- If the interface is connected to a Cisco IP phone, the QoS labels of incoming packets are trusted only when the IP phone is detected.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-3-13

When the Cisco AutoQoS feature is enabled on the first interface, QoS is globally enabled (**mls qos** global configuration command).

When the **auto qos voip trust** interface configuration command is entered, the ingress classification on the interface is set to trust the CoS QoS label received in the packet, and the egress queues on the interface are reconfigured. QoS labels in ingress packets are trusted.

When the **auto qos voip cisco-phone** interface configuration command is entered, the trusted boundary feature is enabled. The trusted boundary feature uses the Cisco Discovery Protocol to detect the presence or absence of a Cisco IP phone. When a Cisco IP phone is detected, the ingress classification on the interface is set to trust the QoS label received in the packet. When a Cisco IP phone is absent, the ingress classification is set to not trust the QoS label in the packet. The egress queues on the interface are also reconfigured. This command extends the trust boundary if an IP phone is detected.

The syntax for the **auto qos voip** command in Cisco IOS Release 12.2(46)SE for the Cisco Catalyst 2960 Switch is as follows:

```
auto qos voip {cisco-phone | cisco-softphone | trust}
```

Syntax Description

Parameter	Description
cisco-phone	Identifies a port as connected to a Cisco IP phone, and automatically configures QoS for VoIP. The QoS labels of incoming packets are trusted only when the telephone is detected.
cisco-softphone	Identifies a port as connected to a device running the Cisco SoftPhone, and automatically configures QoS for VoIP.
trust	Identifies a port as connected to a trusted switch or router, and automatically configures QoS for VoIP. The QoS labels of incoming packets are trusted. For nonrouted ports, the CoS value of the incoming packet is trusted.

Monitoring Cisco AutoQoS VoIP

This topic describes how to use Cisco IOS commands to examine and monitor a network configuration after Cisco AutoQoS has been enabled.

Monitoring Cisco AutoQoS VoIP: Routers

```
router>
```

```
show auto qos [interface interface type]
```

- Displays the interface configurations, policy maps, class maps, and ACLs created on the basis of automatically generated configurations

```
router>show auto qos interface Serial6/0
```

```
Serial6/0 -  
!  
interface Serial6/0  
service-policy output AutoQoS-Policy-UnTrust
```

© 2003 Cisco Systems, Inc. All rights reserved.QoS v2.3-3-14

When the **auto qos voip** command is used to configure the Cisco AutoQoS VoIP feature, configurations are generated for each interface or PVC. These configurations are then used to create the interface configurations, policy maps, class maps, and ACLs. The **show auto qos** command can be used to verify the contents of the interface configurations, policy maps, class maps, and ACLs.

The **show auto qos interface** command can be used with Frame Relay DLCIs and ATM PVCs.

When the **interface** keyword is used along with the corresponding interface type argument, the **show auto qos interface [interface type]** command displays the configurations created by the Cisco AutoQoS VoIP feature on the specified interface.

When the **interface** keyword is used but an interface type is not specified, the **show auto qos interface** command displays the configurations created by the Cisco AutoQoS VoIP feature on all the interfaces or PVCs on which the Cisco AutoQoS VoIP feature is enabled.

The syntax for the **show auto qos** command in Cisco IOS Release 12.4 is as follows:

```
show auto qos [interface [type slot/port]]
```


Syntax Description

Parameter	Description
interface	(Optional) Displays the configurations created by the Cisco AutoQoS VoIP feature on all the interfaces or PVCs on which the Cisco AutoQoS VoIP feature is enabled. When the interface keyword is configured but an interface type is not specified, the show auto qos interface command displays the configurations created by the Cisco AutoQoS VoIP feature on all the interfaces or PVCs on which the Cisco AutoQoS VoIP feature is enabled.
<i>type</i>	(Optional) Specifies an interface type; valid values are atm , ethernet , fastethernet , ge-wan , gigabitethernet , pos , and tengigabitethernet .
<i>slot/port</i>	Module and port number.

Example: show auto qos and show auto qos interface

The **show auto qos** command displays all of the configurations created by the Cisco AutoQoS VoIP feature:

```
Router# show auto qos
Serial6/1.1: DLCI 100 -
!
interface Serial6/1
frame-relay traffic-shaping
!
interface Serial6/1.1 point-to-point
frame-relay interface-dlci 100
class AutoQoS-VoIP-FR-Serial6/1-100
frame-relay ip rtp header-compression
!
map-class frame-relay AutoQoS-VoIP-FR-Serial6/1-100
frame-relay cir 512000
frame-relay bc 5120
frame-relay be 0
frame-relay mincir 512000
service-policy output AutoQoS-Policy-UnTrust
frame-relay fragment 640
```

Monitoring AutoQoS VoIP: Routers (Cont.)

router>

```
show policy-map interface [interface type]
```

- Displays the packet statistics of all classes that are configured for all service policies, on either the specified interface or subinterface

```
router>show policy-map interface FastEthernet0/0.1
FastEthernet0/0.1
Service-policy output: voice_traffic
Class-map: dscp46 (match-any)
0 packets, 0 bytes
5 minute offered rate 0 bps, drop rate 0 bps
Match: ip dscp 46
0 packets, 0 bytes
5 minute rate 0 bps
Traffic Shaping
Target  Byte      Sustain  Excess   Interval  Increment  Adapt
Rate    Limit      bits/int bits/int (ms)      (bytes)    Active
  2500                10000    10000    333        1250         -
.....rest deleted
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-3-15

To display the configuration of all classes configured for all service policies on the specified interface, or to display the classes for the service policy for a specific PVC on the interface, use the **show policy-map interface** EXEC or privileged EXEC command.

The syntax for the **show policy-map** command is as follows:

```
show policy-map interface interface-name [vc [vpi/] vci] [dlsi
dlsi] [input | output]
```

The syntax for the **show policy-map interface** command in Cisco IOS Release 12.4 for the Cisco 3825 Series Integrated Service Router is as follows:

```
show policy-map interface [interface-type interface-number |
null 0 ] [input | output]
```

Syntax Description

Parameter	Description
<i>interface-type</i>	(Optional) Interface type; valid values include Async, BVI, CDMA-Ix, CTunnel, Dialer, GigabitEthernet, Group-Async, Lex, Loopback, MFR, Multilink, Port-channel, Tunnel, Vif, Virtual-PPP, Virtual-Template, Virtual-TokenRing, and XTagATM.
<i>interface-number</i>	(Optional) Module and port number. Valid values depend on the specified interface type.
null 0	(Optional) Specifies the null interface; the only valid value is 0.
input	(Optional) Indicates that the statistics for the attached input policy will be displayed.
output	(Optional) Indicates that the statistics for the attached output policy will be displayed.
Note	Command options vary by router platform. For the command syntax for your router platform, refer to the Cisco IOS Master Command List, Release 12.4 on Cisco.com.

Monitoring AutoQoS VoIP: Switches

Switch#

```
show auto qos [interface interface-id]
```

- Displays the AutoQoS configuration that was initially applied
- Does not display any user changes to the configuration that might be in effect

```
Switch#show auto qos
Initial configuration applied by AutoQoS:
wrr-queue bandwidth 20 1 80 0
no wrr-queue cos-map
wrr-queue cos 1 0 1 2 4
wrr-queue cos 3 3 6 7
wrr-queue cos 4 5
mls qos map cos-dscp 0 8 16 26 32 46 48 56
!
interface FastEthernet0/3
mls qos trust device cisco-phone
mls qos trust cos
```

© 2008 Cisco Systems, Inc. All rights reserved.

QoS v2.2-3-16

To display the initial Cisco AutoQoS configuration, use the **show auto qos [interface [interface-id]]** privileged EXEC command. To display any user changes to that configuration, use the **show running-config** privileged EXEC command. You can compare the **show auto qos** and the **show running-config** command output to identify the user-defined QoS settings.

The syntax for the **show auto qos** command in Cisco IOS Release 12.2(46)SE for the Cisco Catalyst 2960 Switch is as follows:

```
show auto qos [interface [interface-id]]
```

Syntax Description

Parameter	Description
interface [interface-id]	(Optional) Displays auto-QoS information for the specified port or for all ports. Valid interfaces include physical ports.

Monitoring AutoQoS VoIP: Switches (Cont.)

Switch#

```
show mls qos interface [interface-id | vlan vlan-id]
[buffers | policers | queueing | statistics]
[ | {begin | exclude | include} expression]
```

- Displays QoS information at the interface level

```
Switch#show mls qos interface gigabitethernet0/1 statistics
Ingress
  dscp: incoming  no_change  classified  policed  dropped (in bytes)
    1 : 0          0         0           0        0
  Others: 203216935 24234242 178982693 0        0
Egress
  dscp: incoming  no_change  classified  policed  dropped (in bytes)
    1 : 0          n/a       n/a         0        0
WRED drop counts:
  qid      thresh1  thresh2  FreeQ
    1 : 0      0      1024
    2 : 0      0      1024
.....rest deleted
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3--3-17

The **show mls qos interface** command is used to display QoS information at the interface level, including the configuration of the egress queues, the CoS-to-egress-queue map, the interfaces that have configured policers, and ingress and egress statistics (including the number of bytes dropped).

If no keyword is specified with the **show mls qos interface** command, the port QoS mode (DSCP trusted, CoS trusted, untrusted, and so forth), default CoS value, DSCP-to-DSCP-mutation map (if any) that is attached to the port, and policy map (if any) that is attached to the interface, are displayed. If a specific interface is not specified, the information for all interfaces is displayed.

Expressions are case-sensitive. For example, if **exclude output** is entered, the lines that contain *output* are not displayed.

The syntax for the **show mls qos interface** command in Cisco IOS Release 12.2(46)SE for the Cisco Catalyst 2960 switch is as follows:

```
show mls qos interface [interface-id] [buffers | queueing |
statistics] [ | {begin | exclude | include} expression]
```

Syntax Description

Parameter	Description
<i>interface-id</i>	(Optional) Displays QoS information for the specified port. Valid interfaces include physical ports.
<i>buffers</i>	(Optional) Displays the buffer allocation among the queues.
<i>queuing</i>	(Optional) Displays the queuing strategy (shared or shaped) and the weights corresponding to the queues.
<i>statistics</i>	(Optional) Displays statistics for sent and received DSCPs and CoS values, the number of packets enqueued or dropped per egress queue, and the number of in-profile and out-of-profile packets for each policer.
<i>begin</i>	(Optional) Displays begins with the line that matches the expression.
<i>exclude</i>	(Optional) Displays excludes lines that match the expression.
<i>include</i>	(Optional) Displays includes lines that match the specified expression.
<i>expression</i>	Expression in the output to use as a reference point.

Note	Though visible in the command-line help string, the policer keyword is not supported.

Monitoring Cisco AutoQoS VoIP: Switches (Cont.)

Switch#

```
show mls qos maps [cos-dscp | dscp-cos | dscp-mutation  
dscp-mutation-name | dscp-switch-priority | ip-prec-dscp |  
policed-dscp] [ | {begin | exclude | include} expression
```

- Maps are used to generate an internal DSCP value, which represents the priority of the traffic.

```
Switch#show mls qos maps dscp-cos
```

Dscp-cos map:

```
dscp: 0 8 10 16 18 24 26 32 34 40 46 48 56
```

```
-----  
cos: 0 1 1 2 2 3 7 4 4 5 5 6 7
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-3-18

You can use the **show mls qos maps** command to display the current DSCP and CoS mappings.

The syntax for the **show mls qos maps** command in Cisco IOS Release 12.2(46)SE for the Cisco Catalyst 2960 switch is as follows:

```
show mls qos maps [cos-dscp | cos-input-q | cos-output-q |  
dscp-cos | dscp-input-q | dscp-mutation dscp-mutation-name |  
dscp-output-q | ip-prec-dscp | policed-dscp] [ | {begin |  
exclude | include} expression]
```

Syntax Description

Parameter	Description
cos-dscp	(Optional) Displays CoS-to-DSCP map.
cos-input-q	(Optional) Displays the CoS input queue threshold map.
cos-output-q	(Optional) Displays the CoS output queue threshold map.
dscp-cos	(Optional) Displays DSCP-to-CoS map.
dscp-input-q	(Optional) Displays the DSCP input queue threshold map.
<i>dscp-mutation dscp-mutation- name</i>	(Optional) Displays the specified DSCP-to-DSCP-mutation map.
dscp-output-q	(Optional) Displays the DSCP output queue threshold map.
ip-prec-dscp	(Optional) Displays the IP-precedence-to-DSCP map.
policed-dscp	(Optional) Displays the policed-DSCP map.
 begin	(Optional) Displays begins with the line that matches the expression.
 exclude	(Optional) Displays excludes lines that match the expression.
 include	(Optional) Displays includes lines that match the specified expression.
<i>expression</i>	Expression in the output to use as a reference point.

Automation with Cisco AutoQoS

This topic describes several of the QoS technologies that are automatically implemented on the network when using Cisco AutoQoS.

Automation with Cisco AutoQoS (Routers):

DiffServ Functions Automated

DiffServ Function	Cisco IOS/Catalyst Software QoS Feature	Behavior
Classification	NBAR DSCP, Port	Classification of VoIP based on packet attributes or port trust.
Marking	Class-based marking	Set Layer 3 and Layer 2 attributes to categorize packets into a class.
Congestion Management	Percentage-based LLQ, WRR	Provide EF treatment to voice and best-effort treatment to data.
Shaping	Class-based shaping of FRTS	Shape to CIR to prevent burst and smooth traffic to configured rate.
Link Efficiency Mechanism	Header compression	Reduce the VoIP bandwidth requirement.
Link Efficiency Mechanism	Link fragmentation and interleaving	Reduce jitter experienced by voice packets.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS 42.3-3-10

Cisco AutoQoS performs these functions in a WAN:

- Automatically classifies Real-Time Transport Protocol (RTP) payload and VoIP control packets (H.323, H.225 Unicast, Skinny, Session Initiation Protocol (SIP), Media Gateway Control Protocol (MGCP))
- Builds service policies for VoIP traffic that are based on Cisco MQC
- Provisions low-latency queuing (LLQ)—priority queuing (PQ) for VoIP bearer and bandwidth guarantees for control traffic
- Enables WAN traffic shaping that adheres to Cisco best practices, where required
- Enables link efficiency mechanisms such as LFI and cRTP where required
- Provides SNMP and syslog alerts for VoIP packet drops

Cisco AutoQoS performs these functions in a LAN:

- Enforces the trust boundary on Cisco Catalyst switch access ports, and uplinks and downlinks
- Enables Cisco Catalyst strict PQ (also known as expedited queuing) with weighted round-robin (WRR) scheduling for voice and data traffic, where appropriate
- Configures queue admission criteria (maps CoS values in incoming packets to the appropriate queues)
- Modifies queue sizes and weights where required

Summary

This topic summarizes the key points discussed in this lesson.

Summary

- You can enable QoS on a network by a single command per interface using Cisco AutoQoS.
- Cisco AutoQoS is supported on the following routers: Cisco 1760, 1800, 2600 (including XM models), 2800, 3600, 3700, 3800, and 7200 Series Routers.
- Cisco AutoQoS is supported on the following switches: Cisco Catalyst 2950, 2960, 2970, 3550, 3560, 4500, and 6500 Series Switches.
- There are several prerequisites for using Cisco AutoQoS. Cisco AutoQoS automatically configures and enables the DiffServ mechanisms necessary for QoS.
- You can use the **show auto qos** command to verify the contents of the interface configurations, policy maps, class maps, and ACLs.
- Cisco AutoQoS performs several functions on both WAN and LAN.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3—3-20

Introducing Cisco AutoQoS for the Enterprise

Overview

This lesson describes Cisco AutoQoS for the Enterprise and its capabilities and configuration. Cisco AutoQoS for the Enterprise is a process in which two intelligent mechanisms are deployed to detect voice, video, and data traffic in Cisco networks. The mechanisms generate best-practice quality of service (QoS) policies and apply those policies to WAN interfaces. You can enable Cisco AutoQoS for the Enterprise with two simple commands. Cisco AutoQoS for the Enterprise detects up to ten traffic classes and provisions for them.

Objectives

Upon completing this lesson, you will be able to identify capabilities provided by Cisco AutoQoS for the Enterprise and configure QoS on a network using Cisco AutoQoS for the Enterprise. This ability includes being able to meet these objectives:

- Explain how Cisco AutoQoS for the Enterprise is used to implement QoS policy
- Describe the router platforms on which Cisco AutoQoS for the Enterprise is supported
- Describe how Cisco AutoQoS for the Enterprise is configured on a router using CLI
- Describe how to use Cisco IOS commands to examine and monitor a network configuration after Cisco AutoQoS for the Enterprise has been deployed

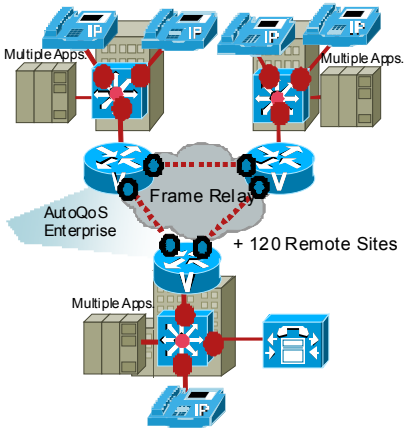
Cisco AutoQoS for the Enterprise

This topic describes how Cisco AutoQoS for the Enterprise is used to implement QoS policy.

Cisco AutoQoS for the Enterprise

- Two commands entered on the WAN interface

```
auto discovery qos [trust]
auto qos
```
- Cisco AutoQoS for the Enterprise:
 - Medium-to-large campus with large remote sites
- Best practices WAN QoS policy automation tool
- Classification for up to 10 traffic classes



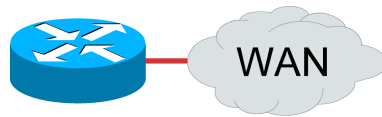
© 2009 Cisco Systems, Inc. All rights reserved. QoS v2.3-3.2

The Cisco AutoQoS for the Enterprise feature automates the deployment of QoS policies in a general business environment. The policies deployed by Cisco AutoQoS for the Enterprise are not solely focused on VoIP convergence, but also on convergence of voice, video, and data applications. Cisco AutoQoS for the Enterprise is generally deployed in midsize companies and branch offices of larger companies. It is used to provide best-practice QoS policy generation for voice as well as to provide for classification of up to ten traffic classes.

Existing QoS policies may be present during the first configuration phase of Cisco AutoQoS for the Enterprise; that is, during the auto-discovery (data collection) phase. However, any existing QoS policies must be removed before the Cisco AutoQoS-generated policies are applied during the second configuration phase of this feature.

Cisco AutoQoS for the Enterprise (Cont.)

AutoDiscovery	Cisco AutoQoS Policy
Application and Protocol Types	Cisco AutoQoS Class Maps Match Statements
Offered Bit Rate (Average and Peak)	Minimum Bandwidth to Class Queues, Scheduling, and WRED



Traffic Class	DSCP
IP Routing	CS6
Interactive Voice	EF
Interactive Video	AF41
Streaming Video	CS4
Telephony Signaling	CS3
Transactional/Interactive	AF21
Network Management	CS2
Bulk Data	AF11
Scavenger	CS1
Best Effort	0

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-3.3

The figure shows the ten traffic classes that can be classified by Cisco AutoQoS for the Enterprise. The auto-discovery process detects applications and protocol types, and offers a recommended QoS policy, which will be generated and installed by the AutoQoS macros, based on the data gathered by the auto-discovery process. The auto-discovery phase gives the discovery period enough time to gather traffic statistics before applying Cisco AutoQoS. The time for gathering traffic statistics before applying Cisco AutoQoS depends upon the actual highs and lows of traffic patterns of a network.

Cisco AutoQoS for the Enterprise (Cont.)

```
class-map match-any AutoQoS-Voice-Se3/1
  match protocol rtp audio
class-map match-any AutoQoS-Inter-Video-Se3/1
  match protocol rtp video
class-map match-any AutoQoS-Signaling-Se3/1
  match protocol sip
  match protocol rtcp
class-map match-any AutoQoS-Transactional-Se3/1
  match protocol citrix
class-map match-any AutoQoS-Bulk-Se3/1
  match protocol exchange
!
policy-map AutoQoS-Policy-Se3/1
  class AutoQoS-Voice-Se3/1
    priority percent 30
    set dscp ef
  class AutoQoS-Inter-Video-Se3/1
    bandwidth remaining percent 1
    set dscp af41
  class AutoQoS-Signaling-Se3/1
    bandwidth remaining percent 5
    set dscp cs3
  class AutoQoS-Transactional-Se3/1
    bandwidth remaining percent 3
    random-detect dscp-based
    set dscp af21
  class AutoQoS-Bulk-Se3/1
    bandwidth remaining percent 1
    random-detect dscp-based
    set dscp af11
  class class-default
    fair-queue
```

← Cisco AutoQoS for the Enterprise

Cisco AutoQoS VoIP

```
!
class-map match-any AutoQoS-VoIP-RTP-Trust
  match ip dscp ef
class-map match-any AutoQoS-VoIP-Control-Trust
  match ip dscp cs3
  match ip dscp af31
!
policy-map AutoQoS-Policy-Trust
  class AutoQoS-VoIP-RTP-Trust
    priority percent 70
  class AutoQoS-VoIP-Control-Trust
    bandwidth percent 5
  class class-default
    fair-queue
!
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-3.4

The figure shows the differences between Cisco AutoQoS for the Enterprise and Cisco AutoQoS VoIP. The Cisco AutoQoS policies came off the same router; Cisco AutoQoS VoIP was first configured and then removed, followed by the configuration of Cisco AutoQoS for the Enterprise. Notice how detailed the Cisco AutoQoS for the Enterprise policy is compared to the Cisco AutoQoS VoIP policy.

The Cisco AutoQoS for the Enterprise policy is based on the traffic statistics gathered by the Cisco AutoQoS discovery process. Conversely, the Cisco AutoQoS VoIP policy was not generated by traffic statistics, but from a generic QoS template.

Cisco AutoQoS for the Enterprise Router Platforms

This topic describes the router platforms on which Cisco AutoQoS can be used.

Cisco AutoQoS for the Enterprise: Router Platforms

Cisco AutoQoS for the Enterprise supported platforms:

- Cisco 1760 Modular Access Routers
- Cisco 1800 Series Integrated Services Routers
- Cisco 2600 Series Multiservice Platforms, including XM Series
- Cisco 2800 Series Integrated Services Routers
- Cisco 3600 Series Multiservice Platforms
- Cisco 3725 and 3745 Multiservice Access Routers
- Cisco 3800 Series Integrated Services Routers
- Cisco 7200 Series Routers

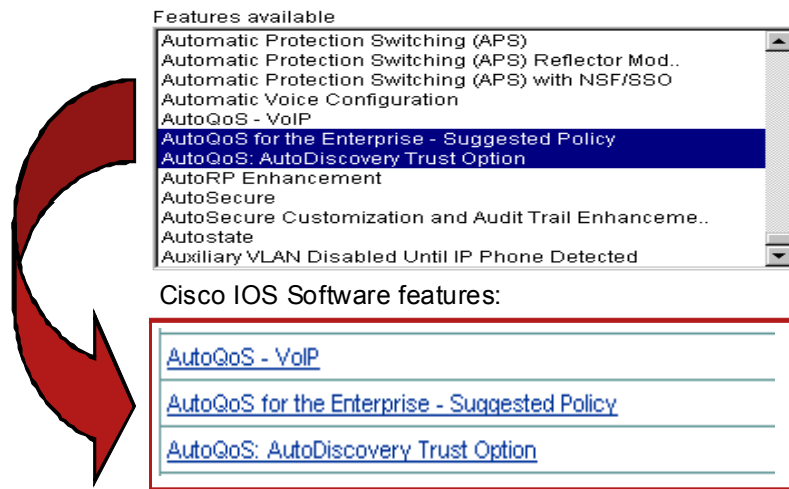
© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-3-5

The figure shows the listed Cisco router platforms that support the Cisco AutoQoS for the Enterprise feature. When verifying whether a Cisco IOS Software release supports Cisco AutoQoS for the Enterprise, check the Cisco Feature Navigator tool on <http://www.cisco.com>.

Cisco AutoQoS for the Enterprise: Router Platforms (Cont.)

Search Cisco IOS Software by FEATURE:



© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-3.6

When searching for a Cisco IOS Software version that will support the Cisco AutoQoS for the Enterprise feature, make sure, when using the Cisco IOS Feature Navigator tool, that you select these two options, as indicated in the “Features available” list shown in the figure:

- **AutoQoS for the Enterprise - Suggested Policy:** Discovers and automatically configures QoS policies for application traffic.
- **AutoQoS: AutoDiscovery Trust Option:** Extends the power of Cisco AutoQoS for the Enterprise to those routers on which differentiated services code point (DSCP) values have already been assigned to traffic at the network edge. This option enables customers to automatically set the QoS policy on routers by allowing the network to trust internally established priority levels for various types of traffic.

Configuring Cisco AutoQoS for the Enterprise

This topic describes the prerequisites for configuring Cisco AutoQoS for the Enterprise.

Configuring Cisco AutoQoS for the Enterprise

- Before configuring Cisco AutoQoS for the Enterprise, remove any existing service policies on an interface.
- Cisco AutoQoS for the Enterprise requires Cisco Express Forwarding to be enabled at the interface level.
- The correct bandwidth must be configured on all interfaces or subinterfaces using the **bandwidth** command.
- If the interface or subinterface has a link speed of 768 kb/s or lower, an IP address must be configured using the **ip address** command.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS 12.3-3.7

Before configuring Cisco AutoQoS on a Cisco router, you must meet these prerequisites:

- Removal of any preexisting service policies.
- Cisco Express Forwarding must be enabled at the interface level on ATM permanent virtual circuit (PVC) or Frame Relay data-link connection identifiers (DLCIs) (point-to-point subinterfaces only).
- Cisco AutoQoS uses network-based application recognition (NBAR) to identify various applications and traffic types, and Cisco Express Forwarding is a prerequisite for NBAR.
- Cisco AutoQoS classifies links as either low-speed or high-speed, depending upon the link bandwidth. If the default bandwidth is not specified on a serial connection, it is 1.544 Mb/s. You must specify the correct bandwidth on the interface or subinterface where Cisco AutoQoS is to be enabled, as follows:
 - For all interfaces or subinterfaces, be sure to properly configure the bandwidth by using the **bandwidth** command. The amount of bandwidth that is allocated should be based on the link speed of the interface.
 - If the interface or subinterface has a link speed of 768 kb/s or lower, an IP address must be configured on the interface or subinterface using the **ip address** command. By default, Cisco AutoQoS will enable Multilink PPP (MLP) and copy the configured IP address to the multilink bundle interface.

In addition to the Cisco AutoQoS prerequisites, there are other recommendations and requirements when configuring Cisco AutoQoS, as follows: (Be aware that these may change with Cisco IOS Software releases and should be verified before implementing Cisco AutoQoS in the environment.)

- The Cisco AutoQoS for the Enterprise feature is supported only on the following interfaces and PVCs:
 - ATM PVCs low-speed and high-speed subinterfaces only
 - Serial interfaces with PPP or High-Level Data Link Control (HDLC)
 - Frame Relay DLCIs (point-to-point subinterfaces only)Cisco AutoQoS does not support Frame Relay multipoint interfaces.
- You can manually tune a configuration template, generated by configuring Cisco AutoQoS on an interface or PVC, via a command-line interface (CLI), if desired.
- Cisco AutoQoS cannot be configured if a QoS service policy is already configured and attached to the interface or PVC.
- MLP is configured automatically for a serial interface with a low-speed link when Real-Time Transport Protocol (RTP) traffic is detected. The serial interface must have an IP address, which is removed and put on the MLP bundle. Cisco AutoQoS for the Enterprise must also be configured on the other side of the link.
- The **no auto discovery qos** command removes auto discovery on WAN interfaces.
- The **no auto qos voip** command removes Cisco AutoQoS VoIP. However, if the interface or PVC on which Cisco AutoQoS-generated QoS configuration is deleted without configuring the **no auto qos voip** command, Cisco AutoQoS VoIP will not be completely removed from the configuration.
- Cisco AutoQoS Simple Network Management Protocol (SNMP) traps are only delivered when an SNMP server is used in conjunction with Cisco AutoQoS.
- The SNMP community string “AutoQoS” should have “write” permission.
- If the device is reloaded with the saved configuration after configuring Cisco AutoQoS and saving the configuration to NVRAM, some warning messages may be generated by Remote Monitoring (RMON) threshold commands. You can ignore these warning messages. (To avoid further warning messages, save the configuration to NVRAM again without making any changes to the QoS configuration.)
- By default, Cisco 7200 Series Routers and below that support Modular QoS CLI (MQC) reserve up to 75 percent of the interface bandwidth for user-defined classes. The remaining bandwidth is used for the default class. However, the entire remaining bandwidth is *not* guaranteed to the default class. This bandwidth is shared proportionally between the different flows in the default class and excess traffic from other bandwidth classes. By default, at least one percent of the available bandwidth is reserved and guaranteed for class default traffic on Cisco 7500 Series Routers. (You can allocate up to 99 percent to the other classes.)

Configuring Cisco AutoQoS for the Enterprise (Cont.)

```
router(config-if)# or router(config-fr-dlci)#
```

```
auto discovery qos [trust]
```

- Configures phase one of the auto-discovery phase
- Traffic statistics are gathered by NBAR

```
router(config-if)# or router(config-fr-dlci)#
```

```
auto qos
```

- This command will not work until statistics have been gathered from phase one.
- Configures the second phase for Cisco AutoQoS for the Enterprise.
- Instructs the templates to generate QoS policy based on statistics gathered using NBAR and installs the service policies in the WAN interface where the command is entered.

© 2009 Cisco Systems, Inc. All rights reserved.QoS v2.3-3.8

To configure Cisco AutoQoS for the Enterprise, enter **auto discovery qos [trust]** first, and then, after two to three days and after the verification of traffic statistics, add the **auto qos** commands on a router WAN interface. To verify that traffic has been gathered, enter **show auto discovery qos [interface]**. The router will not accept the **auto qos** command until statistics have been gathered.

Note Configuring **auto qos voip** along with **auto discovery qos [trust]** defaults the Cisco AutoQoS policy generation to Cisco AutoQoS VoIP.

To remove the Cisco AutoQoS discovery or Cisco AutoQoS VoIP feature from an interface, use the **no** form of the **auto discovery qos** or **auto qos voip** commands:

The syntax for the **auto discovery qos** command is as follows:

```
auto discovery qos [trust]
```

Syntax Description

Parameter	Description
trust	(Optional) Specifies that the DSCP markings of a packet are trusted; that is, used for classification of the voice, video, and data traffic.

Example: Configuring the Cisco AutoQoS Discovery Feature on a High-Speed Serial Interface

In the example, the Cisco AutoQoS discovery is configured on the high-speed serial interface s1/1:

```
Router> enable
Router# configure terminal
Router(config)# interface s1/1
Router(config-if)# bandwidth 1540
Router(config-if)# ip address 10.10.100.1 255.255.255.0
Router(config-if)# auto discovery qos
Router(config-if)# exit
```

Example: Configuring the Cisco AutoQoS for the Enterprise Feature on a High-Speed Serial Interface

In the example, the Cisco AutoQoS for the Enterprise feature is configured on the high-speed serial interface s1/1:

```
Router> enable
Router# configure terminal
Router(config)# interface s1/1
Router(config-if)# bandwidth 1540
Router(config-if)# ip address 10.10.100.1 255.255.255.0
Router(config-if)# auto discovery qos
Router(config-if)# auto qos
Router(config-if)# exit
```

Monitoring Cisco AutoQoS for the Enterprise

This topic describes how to use Cisco IOS commands to examine and monitor a network configuration after Cisco AutoQoS has been configured.

Monitoring Cisco AutoQoS for the Enterprise

```
router>
```

```
show auto qos [interface interface type]
```

- Displays the interface configurations, policy maps, class maps, and ACLs created on the basis of automatically generated configurations

```
router>show auto qos interface Serial2/1
```

```
Serial2/1 -  
!  
interface Serial2/1.1  
service-policy output AutoQoS-Policy-Trust
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-3.8

When the Cisco AutoQoS for the Enterprise feature is configured on a router WAN interface, you can use the **show auto qos** command to verify the contents of the interface configurations, policy maps, class maps, and ACLs.

You can use the **show auto qos interface** command with Frame Relay DLCIs and ATM PVCs. The **show auto qos interface [interface type]** command displays the configurations created by the AutoQoS VoIP feature on the specified interface.

When the **interface** keyword is used but an interface type is not specified, the **show auto qos interface** command displays the configurations created by the Cisco AutoQoS for the Enterprise feature on all the interfaces or PVCs on which the Cisco AutoQoS for the Enterprise feature is enabled.

The syntax for the **show auto qos** command in Cisco IOS Software Release 12.4 is as follows:

```
show auto qos [interface [type slot/port]]
```

Syntax Description

Parameter	Description
interface	(Optional) Displays the configurations created by the Cisco AutoQoS VoIP feature on all the interfaces or PVCs on which the Cisco AutoQoS VoIP feature is enabled. When the interface keyword is configured but an interface type is not specified, the show auto qos interface command displays the configurations created by the Cisco AutoQoS VoIP feature on all the interfaces or PVCs on which the Cisco AutoQoS VoIP feature is enabled.
<i>type</i>	(Optional) Specifies an interface type; valid values are atm , ethernet , fastethernet , ge-wan , gigabitethernet , pos , and tengigabitethernet .
<i>slot/port</i>	Module and port number.

Example: show auto qos and show auto qos interface

In the example, the **show auto qos** command displays all of the configurations created by the Cisco AutoQoS VoIP feature:

```
Router# show auto qos
Serial2/1.1: DLCI 100 -
!
interface Serial2/1
frame-relay traffic-shaping
!
interface Serial2/1.1 point-to-point
frame-relay interface-dlci 100
class AutoQoS-VoIP-FR-Serial2/1.1-100
frame-relay ip rtp header-compression
!
map-class frame-relay AutoQoS-VoIP-FR-Serial2/1.1-100
frame-relay cir 512000
frame-relay bc 5120
frame-relay be 0
frame-relay mincir 512000
service-policy output AutoQoS-Policy-UnTrust
frame-relay fragment 640
```

Monitoring Cisco AutoQoS for the Enterprise (Cont.)

```
Router# show auto discovery qos
AutoQoS Discovery enabled for applications
Discovery up time: 2 days, 55 minutes
AutoQoS Class information:
Class VoIP:
Recommended Minimum Bandwidth: 517 Kbps/50% (PeakRate)
Detected applications and data:
Application/      AverageRate      PeakRate      Total
Protocol          (kbps/%)        (kbps/%)      (bytes)
rtp audio         76/7            517/50        703104
Class Interactive Video:
Recommended Minimum Bandwidth: 24 Kbps/2% (AverageRate)
Detected applications and data:
Application/      AverageRate      PeakRate      Total
Protocol          (kbps/%)        (kbps/%)      (bytes)
rtp video         24/2            5337/52       704574
Class Transactional:
Recommended Minimum Bandwidth: 0 Kbps/0% (AverageRate)
Detected applications and data:
Application/      AverageRate      PeakRate      Total
Protocol          (kbps/%)        (kbps/%)      (bytes)
citrix            36/3            74/7          30212
sqlnet            12/1            7/<1          1540
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3--3-10

The figure shows an example of NBAR recognizing traffic classes that have been detected across a WAN interface. This is only shown in this format after you enter the **auto discovery qos** command on a WAN interface and traffic has been detected. The next step would be to enter Cisco AutoQoS configuration under the WAN interface, which would result in automatically configuring a best-practice QoS policy set for the supported traffic.

Monitoring Cisco AutoQoS for the Enterprise (Cont.)

- Suggested policy is based on autodiscovery statistics (NBAR)

```
Router# show auto qos
!
policy-map AutoQoS-Policy-Se2/1.1
class AutoQoS-Voice-Se2/1.1
  priority percent 33
  set dscp ef
class AutoQoS-Inter-Video-Se2/1.1
  bandwidth remaining percent 1
  set dscp af41
class AutoQoS-Signaling-Se2/1.1
  bandwidth remaining percent 1
  set dscp cs3
class AutoQoS-Transactional-Se2/1.1
  bandwidth remaining percent 1
  random-detect dscp-based
  set dscp af21
class AutoQoS-Bulk-Se2/1.1
  bandwidth remaining percent 1
  random-detect dscp-based
  set dscp af11
class class-default
  fair-queue
```

```
Suggested AutoQoS Policy for the current uptime:
!
class-map match-any AutoQoS-Voice-Se2/1.1
  match protocol rtp audio
!
class-map match-any AutoQoS-Inter-Video-Se2/1.1
  match protocol rtp video
!
class-map match-any AutoQoS-Signaling-Se2/1.1
  match protocol sip
  match protocol rtcp
!
class-map match-any AutoQoS-Transactional-Se2/1.1
  match protocol citrix
!
class-map match-any AutoQoS-Bulk-Se2/1.1
  match protocol exchange
!
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-3-11

The figure shows an example of a suggested QoS policy configuration generated by Cisco AutoQoS templates, based on NBAR statistics gathered during the data discovery phase. Notice that these policies are structured as if the policy was configured using the MQC. You can modify these policies.

Summary

This topic summarizes the key points discussed in this lesson.

Summary

- Cisco AutoQoS for the Enterprise is used on router WAN interfaces and applies best-practice QoS automation. Ten traffic types are detected.
- Cisco AutoQoS for the Enterprise is supported on Cisco 1700, 2600 (including XM models), 2800, 3600, 3700, 3800, and 7200 Series Routers.
- Cisco AutoQoS for the Enterprise is enabled by two commands. (**auto discovery** and **auto qos**).
- After the Cisco AutoQoS for the Enterprise feature is configured on a router WAN interface, you can use the **show auto qos** command to verify the contents of the interface configurations, policy maps, class maps, and ACLs.

Module Summary

This topic summarizes the key points discussed in this module.

Module Summary

- Modular QoS is a three-step building-block approach to implementing QoS in a network.
- Each class of traffic is defined in a class-map module.
- A policy map module defines a traffic policy that configures the QoS features associated with a traffic class, previously identified using a class map.
- A service policy attaches a traffic policy configured with a policy map to an interface.
- Cisco AutoQoS can be enabled on a network by entering one or two commands on an interface.
- Cisco AutoQoS VoIP works on a variety of Cisco routers and switches and automatically configures and enables the mechanisms necessary for deploying QoS for VoIP.
- Cisco AutoQoS for the Enterprise works on a variety of Cisco routers and automatically configures and enables the mechanisms necessary for deploying up to 10 traffic classes, including VoIP.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-3.1

Both the Modular QoS command-line interface (CLI), or MQC, and Cisco AutoQoS are designed to aid in more rapid and consistent design, implementation, and maintenance of QoS policies for converged networks. The MQC offers a three-step, building-block approach for implementing extremely modular QoS policies for network administrators who have the responsibility to manage large and complex networks. Cisco AutoQoS provides an easy-to-use, mostly automated means to provide consistent QoS policies throughout a network with a minimal design and implementation effort.

References

For additional information, refer to these resources:

- For more information on MQC, refer to *Modular Quality of Service Command-Line Interface Overview* at this URL:
http://www.cisco.com/en/US/docs/ios/12_2/qos/configuration/guide/qcfmdcli.html.
- For more information on MQC, refer to the *Applying QoS Features Using the MQC* section of the *Cisco IOS Quality of Service Solutions Configuration Guide, Release 12.4* at this URL:
http://www.cisco.com/en/US/docs/ios/qos/configuration/guide/qos_mqc_ps6350_TSD_Products_Configuration_Guide_Chapter.html.
- For more information on Cisco AutoQoS, refer to *Cisco AutoQoS Whitepaper* at this URL:
http://www.cisco.com/en/US/tech/tk543/tk759/technologies_white_paper09186a00801348bc.shtml.
- For more information on Cisco AutoQoS, refer to the *Configuring Automatic QoS* section of the *Catalyst 6500 Series Software Configuration Guide, 8.7* at this URL:
<http://www.cisco.com/en/US/partner/docs/switches/lan/catalyst6500/catos/8.x/configuration/guide/autoqos.html>.
- For more information on Cisco AutoQoS, refer to the *AutoQoS – VoIP* section of the *Cisco IOS Quality of Service Solutions Configuration Guide, Release 12.4* at this URL:
http://www.cisco.com/en/US/partner/docs/ios/qos/configuration/guide/autoqos_voip_ps6350_TSD_Products_Configuration_Guide_Chapter.html.

Module Self-Check

Use the questions here to review what you learned in this module. The correct answers and solutions are found in the Module Self-Check Answer Key.

- Q1) Which command would you use to attach a QoS policy to an interface? (Source: Introducing MQC)
- A) **policy-set-interface**
 - B) **policy-map**
 - C) **policy-interface**
 - D) **service-policy**
- Q2) What is “trusted” when the **auto qos voip** command is configured with the “trust” parameter on routers? (Source: Introducing Cisco AutoQoS VoIP)
- A) source address
 - B) MAC address of sender
 - C) DES keyword
 - D) DSCP
- Q3) Which of these terms are displayed by the **show auto qos interface S0/0** router command? (Source: Introducing Cisco AutoQoS VoIP)
- A) ACL configuration applied to the S0/0 interface
 - B) class-maps configuration
 - C) policy-maps configuration
 - D) service-policy configuration on the S0/0 interface
- Q4) Which command would you use on a Cisco IOS-based Catalyst switch to display the configuration of the egress queues? (Source: Introducing Cisco AutoQoS VoIP)
- A) **show mls qos maps**
 - B) **show auto qos**
 - C) **show auto qos interface**
 - D) **show mls qos interface**
- Q5) Which three of these does Cisco AutoQoS VoIP do when used to automatically configure a WAN interface? (Choose three.) (Source: Introducing Cisco AutoQoS for the Enterprise)
- A) enable payload compression
 - B) provision low-latency queuing (LLQ)
 - C) automatically classify RTP payload and VoIP control packets
 - D) enable LFI where required

Module Assessment Answer Key

- Q1) D
- Q2) D
- Q3) D
- Q4) D
- Q5) B, C, D

Classification and Marking

Overview

In any network in which networked applications require differentiated levels of service, traffic must be sorted into different classes upon which quality of service (QoS) is applied. Classification and marking are two critical functions of any successful QoS implementation. Classification allows network devices to identify traffic as belonging to a specific class with specific QoS requirements as determined by an administrative QoS policy. After network traffic is sorted, individual packets are colored or marked so that other network devices can apply QoS features uniformly to those packets in compliance with the defined QoS policy. This module introduces classification and marking and the different methods of performing these critical QoS functions on Cisco routers and switches.

Because the end result of QoS implementations is to affect application traffic traversing a QoS-enabled network, this module also describes the effects that classification and marking have on network traffic.

Module Objectives

Upon completing this module, you will be able to successfully classify and mark network traffic to implement a policy according to QoS requirements. This ability includes being able to meet these objectives:

- Explain the purpose of classification and marking and how they can be used to define a QoS service class
- Define a QoS policy that assigns network traffic to service classes and assigns where classification and marking should be applied to the network
- Use MQC commands to classify packets
- Use class-based marking to assign packets to a specific service class
- Use NBAR to discover network protocols and applications, and to classify packets
- Use the QoS preclassify feature to classify GRE, IPsec, L2F, and L2TP encapsulated packets
- Explain how to implement classification and marking in an interdomain network using QPPB

- Describe LAN-based methods for implementing classification and marking
- Identify the QoS status of packets as they pass through various points in a network

Understanding Classification and Marking

Overview

Quality of service (QoS) offers the ability to provide different levels of treatment to specific classes of traffic. Before any QoS applications or mechanisms can be applied, traffic must be identified and sorted into different classes. QoS is applied to these different traffic classes. Network devices use classification to identify traffic as belonging to a specific class. After network traffic is sorted, marking can be used to color (tag) individual packets so that other network devices can apply QoS features uniformly to those packets as they travel through the network. This lesson introduces the concepts of classification and marking, explains the different markers that are available at the data-link and network layers, and identifies where classification and marking should be used in a network. In addition, the concept of a QoS service class and how a service class can be used to represent an application or set of applications is discussed.

Objectives

Upon completing this lesson, you will be able to explain the purpose of classification and marking and how they can be used to define a QoS service class. This ability includes being able to meet these objectives:

- Explain the purpose of packet classification
- Explain the purpose of packet marking
- Describe IP packet classification and marking at the data-link layer
- Describe IP packet classification and marking at the network layer
- Describe data-link-to-network layer interoperability between QoS markers
- Explain the term “QoS service class” and describe how service classes can be used to create a service policy throughout a network
- Explain how link-layer and network-layer markings are used to define service classes and describe the different applications represented by each of these service classes
- Explain the concept of trust boundaries and how they are used with classification and marking

Classification

This topic describes the purpose of packet classification.

Classification

- The component of a QoS feature that recognizes and distinguishes between different traffic streams
- Most fundamental QoS building block
- Without classification, all packets treated the same

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.2

Classification is the process of identifying traffic and categorizing that traffic into different classes. Packet classification uses a traffic descriptor to categorize a packet within a specific group in order to define that packet. Typically used traffic descriptors include class of service (CoS), incoming interface, IP precedence, differentiated services code point (DSCP), source or destination address, application, and Multiprotocol Label Switching (MPLS) experimental bits (EXP). After the packet has been defined (that is, classified), the packet is then accessible for QoS handling on the network.

Using packet classification, you can partition network traffic into multiple priority levels or classes of service. When traffic descriptors are used to classify traffic, the source agrees to adhere to the contracted terms and the network promises a QoS. Different QoS mechanisms, such as traffic policing, traffic shaping, and queuing techniques use the traffic descriptor of the packet (that is, the classification of the packet) to ensure adherence to that agreement.

Classification should take place at the network edge, typically in the wiring closet, in IP phones or at network endpoints. It is recommended that classification occur as close to the source of the traffic as possible.

Marking

This topic describes the purpose of packet marking.

Marking

- The QoS feature component that “colors” a packet (frame) so that it can be identified and distinguished from other packets (frames) in QoS treatment.
- Commonly used markers include: CoS (ISL, 802.1p), DSCP, IP precedence, and MPLS experimental bits.

© 2009 Cisco Systems, Inc. All rights reserved.

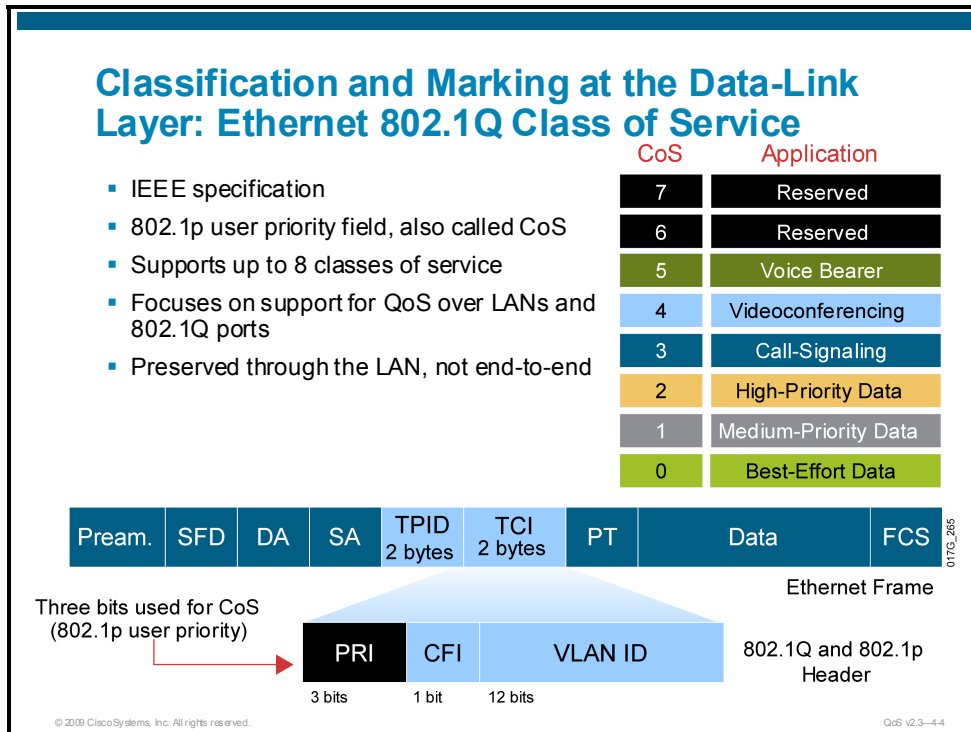
QoS v2.3-4.3

Marking is related to classification. Marking allows network devices to classify a packet or frame based on a specific traffic descriptor. Typically used traffic descriptors include CoS, DSCP, IP precedence, and MPLS experimental bits. Marking can be used to set information in the Layer 2 or Layer 3 packet headers.

Marking a packet or frame with its classification allows network devices to easily distinguish the marked packet or frame as belonging to a specific class. After the packets or frames are identified as belonging to a specific class, QoS mechanisms can be uniformly applied to ensure compliance with administrative QoS policies.

Classification and Marking at the Data-Link Layer

This topic describes IP packet classification and marking options that are available at the data-link layer.



The 802.1Q standard is an IEEE specification for implementing VLANs in Layer 2 switched networks. The 802.1Q specification defines two 2-byte fields, Tag Protocol Identifier (TPID) and Tag Control Information (TCI), which are inserted within an Ethernet frame following the source address field. The TPID field is currently fixed and assigned the value 0x8100. The TCI field is composed of these three fields:

- **User priority bits (3 bits):** The specifications of this 3-bit field are defined by the IEEE 802.1p standard. These bits can be used to mark packets as belonging to a specific CoS. The CoS marking uses the three 802.1p user priority bits and allows a Layer 2 Ethernet frame to be marked with eight different levels of priority (values 0–7). Three bits allow for eight levels of classification, allowing a direct correspondence with IPv4 (IP precedence) type of service (ToS) values. The 802.1P specification defines these standard definitions for each CoS:
 - CoS 7 (111): network
 - CoS 6 (110): Internet
 - CoS 5 (101): critical
 - CoS 4 (100): flash-override
 - CoS 3 (011): flash
 - CoS 2 (010): immediate
 - CoS 1 (001): priority
 - CoS 0 (000): routine

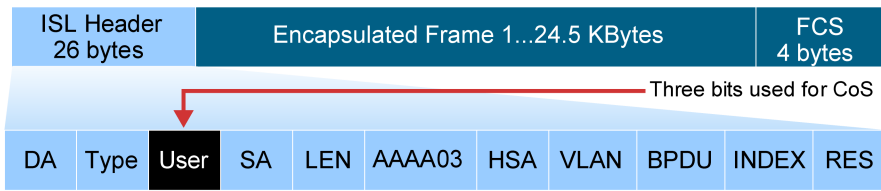
One disadvantage of using CoS markings is that frames will lose their CoS markings when transiting a non-802.1Q or non-802.1p link, including any type of non-Ethernet WAN link. Therefore, a more permanent marking should be used for network transit, such as Layer 3 IP DSCP marking. This is typically accomplished by translating a CoS marking into another marker or simply using a different marking mechanism.

- **Canonical format identifier (CFI) (1 bit):** This bit indicates whether the bit order is canonical or noncanonical. The CFI bit is used for compatibility between Ethernet and Token Ring networks.
- **VLAN identifier (VLAN ID) (12 bits):** The VLAN ID field is a 12-bit field that defines the VLAN used by 802.1Q. The fact that the field is 12 bits restricts the number of VLANs supported by 802.1Q to 4096. For most enterprise customers, 4096 VLANs is adequate. For service provider applications, 4096 VLANs may not be enough.

Classification and Marking at the Data-Link Layer: Cisco ISL Class of Service

- Cisco proprietary specification
- ISL encapsulation adds 30 bytes to Ethernet frame
- ISL header contains VLAN field and User field
- VLAN field contains VLAN ID
- User field carries CoS values
- Supports up to 8 classes of service
- Focuses on support for QoS over ISL trunks
- Preserved through the LAN, not end to end

CoS	Application
7	Reserved
6	Reserved
5	Voice Bearer
4	Videoconferencing
3	Call-Signaling
2	High-Priority Data
1	Medium-Priority Data
0	Best-Effort Data



© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.5

Inter-Switch Link (ISL) is a proprietary Cisco protocol for interconnecting multiple switches and maintaining VLAN information as traffic travels between switches. ISL was created prior to the standardization of 802.1Q. However, ISL is compliant with the IEEE 802.1P standard.

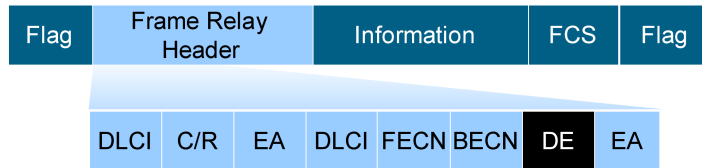
The ISL frame header contains a 4-bit User field that carries 802.1p CoS values in the three least significant bits. When an ISL frame is marked for priority, the three 802.1p CoS bits are set to a value 0–7. In compliance with the 802.1p specification, ISL follows the standard definitions for each CoS:

- CoS 7 (111): network
- CoS 6 (110): Internet
- CoS 5 (101): critical
- CoS 4 (100): flash-override
- CoS 3 (011): flash
- CoS 2 (010): immediate
- CoS 1 (001): priority
- CoS 0 (000): routine

Similar to 802.1Q, ISL CoS markings are not maintained end-to-end if a non-ISL or non-802.1Q trunk is transited. As a result, network administrators typically translate CoS markings into another marker or simply use a different marking mechanism altogether.

Classification and Marking at the Data-Link Layer: Frame Relay and ATM QoS

Frame Relay Frame



- Frame Relay DTE devices can set the DE bit of a frame so that if the network becomes congested, Frame Relay devices will discard frames with the DE bit set before discarding those that do not have the DE bit set.
- Preserved throughout the Frame Relay network.

ATM UNI cell



- The CLP bit indicates that the cell should be discarded if it encounters congestion as it moves through the network.
- Preserved throughout the ATM network.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.6

Before the Internet Engineering Task Force (IETF) defined QoS methods for the network layer, the ITU-T, ATM Forum, and the Frame Relay Forum (FRF) had already derived standards for link-layer QoS in ATM and Frame Relay networks.

ATM standards define a rich QoS infrastructure by supporting traffic contracts, many adjustable QoS knobs (such as peak cell rate [PCR], minimum cell rate [MCR], and so on), signaling, and admission control. Frame Relay provides a simpler set of QoS mechanisms to ensure a committed information rate (CIR), congestion notification, and FRF.12.

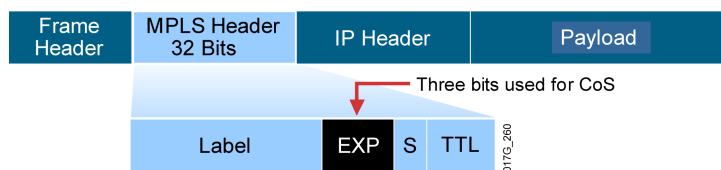
One component of Frame Relay QoS is packet discard when congestion is experienced in the network. Frame Relay will allow network traffic to be sent at a rate exceeding its CIR. Frames sent that exceed the committed rate can be marked as discard eligible (DE). If congestion occurs in the network, frames marked DE will be discarded prior to frames that are not marked.

ATM cells consist of 48 bytes of payload and 5 bytes of header. The ATM header includes the 1-bit cell loss priority (CLP) field, which indicates the drop priority of the cell if that cell encounters extreme congestion as it moves through the ATM network. The CLP bit represents two values: 0 to indicate higher priority and 1 to indicate lower priority. Setting the CLP bit to 1 lowers the priority of the cell, increasing the likelihood that the cell will be dropped when the ATM network experiences congestion.

Classification and Marking at the Data-Link Layer: MPLS Experimental Bits

- MPLS uses a 32-bit label field (shim header) which is inserted between Layer 2 and Layer 3 headers (frame mode).
- Supports up to 8 classes of service.
- The IP precedence or DSCP field is not directly visible to MPLS label switch routers.
- By default, Cisco IOS Software copies the three most significant bits of the DSCP or the IP precedence of the IP packet to the EXP field.
- Preserved throughout the MPLS network.

EXP	Application (DSCP)
7	Reserved CS7
6	Reserved CS6
5	EF
4	AF4x
3	AF3x
2	AF2x
1	AF1x
0	Default



© 2009 Cisco Systems, Inc. All rights reserved.

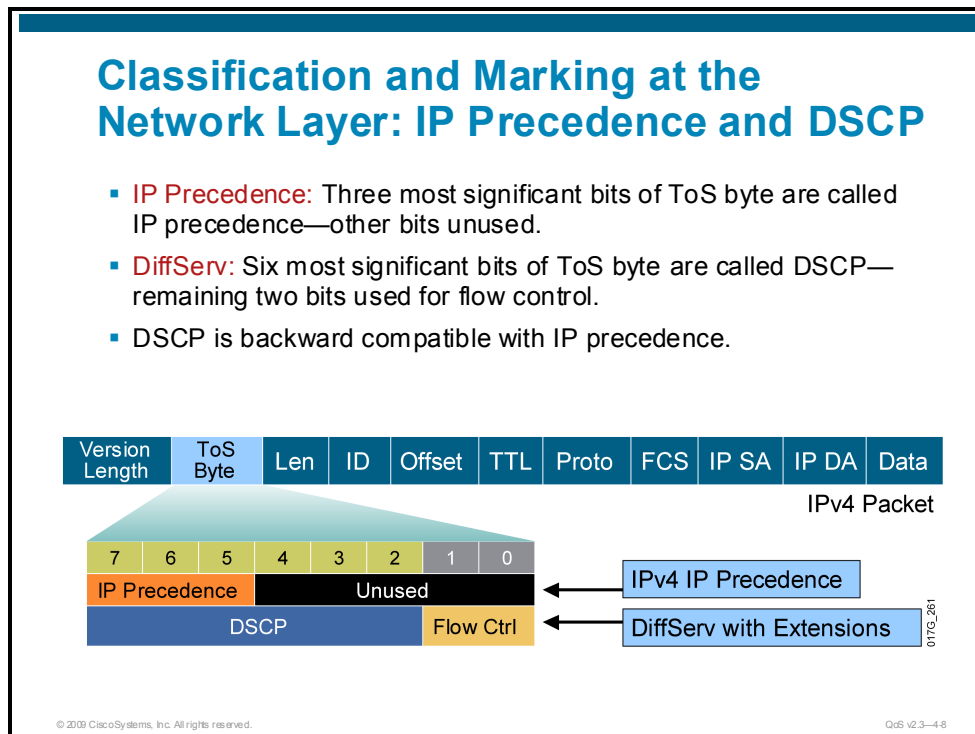
QoS v2.3-4.7

When a customer transmits IP packets from one site to another, the IP precedence field (the first three bits of the DSCP field in the header of an IP packet) specifies the CoS. Based on the IP precedence marking, the packet is given the desired treatment, such as guaranteed bandwidth or latency. If the service provider network is an MPLS network, the IP precedence bits are copied into the MPLS experimental field at the edge of the network. However, the service provider might want to set an MPLS packet QoS to a different value that is determined by the service offering.

The MPLS experimental field allows the service provider to provide QoS without overwriting the value in the customer IP precedence field. The IP header remains available for customer use, and the IP packet marking is not changed as the packet travels through the MPLS network.

Classification and Marking at the Network Layer

This topic describes the different IP packet classification and marking options that are available at the network layer.



At the network layer, IP packets are typically classified based on source or destination IP address, packet length, or the contents of the ToS byte. Link-layer media often changes as a packet travels from its source to its destination. Because a CoS field does not exist in a standard Ethernet frame, CoS markings at the link layer are not preserved as packets traverse nontrunked or non-Ethernet networks. Using marking at the network layer (Layer 3) provides a more permanent marker that is preserved from source to destination.

Originally, only the first three bits of the ToS byte were used for marking, referred to as IP precedence. However, newer standards have made the use of IP precedence obsolete in favor of using the first six bits of the ToS byte for marking, referred to as DSCP.

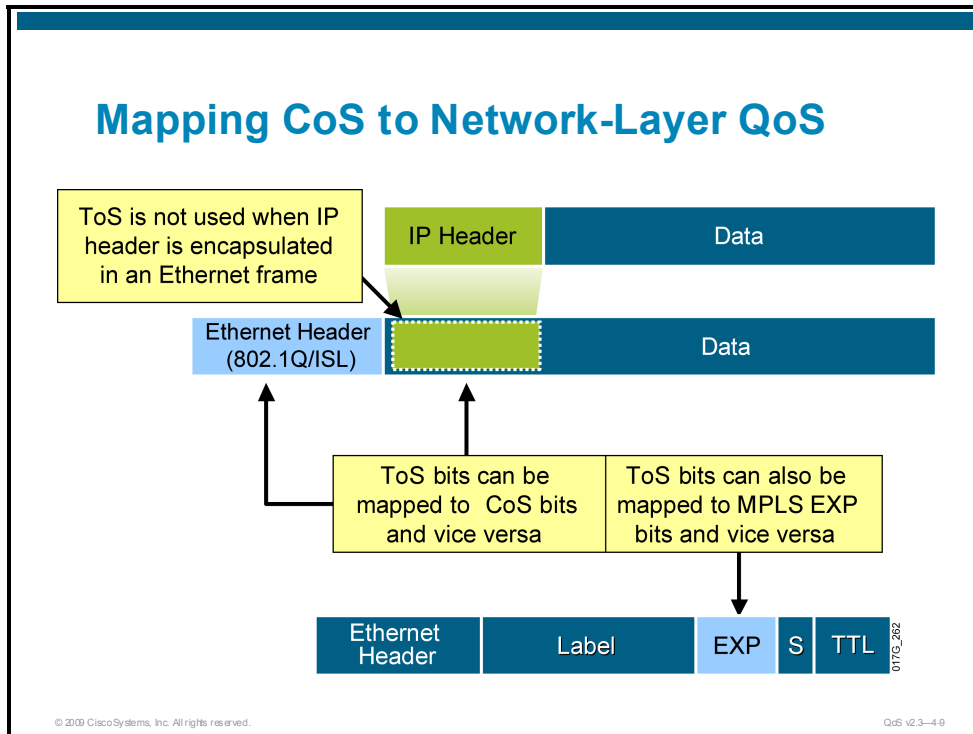
The header of an IPv4 packet contains the ToS byte. IP precedence uses three precedence bits in the ToS field of the IPv4 header to specify CoS for each packet. IP precedence values range from 0 to 7 and allow you to partition traffic in up to six useable classes of service. (Settings 6 and 7 are reserved for internal network use.)

Differentiated services (DiffServ) is a new model that supersedes—and is backward compatible with—IP precedence. DiffServ redefines the ToS byte as the DiffServ field and uses six prioritization bits that permit classification of up to 64 values (0 to 63), of which 32 are commonly used. A DiffServ value is called a DSCP.

With DiffServ, packet classification is used to categorize network traffic into multiple priority levels or classes of service. Packet classification uses the DSCP traffic descriptor to categorize a packet within a specific group to define that packet. After the packet has been defined (classified), the packet is then accessible for QoS handling on the network.

Mapping CoS to Network Layer QoS

This topic describes data-link-layer to network-layer interoperability between different QoS markers.



IP headers are preserved end-to-end when IP packets are transported across a network; data-link-layer headers are not preserved. This means that the IP layer is the most logical place to mark packets for end-to-end QoS. However, there are edge devices that can only mark frames at the data-link layer, and there are many other network devices that only operate at the data-link layer. To provide true end-to-end QoS, the ability to map QoS marking between the data-link layer and the network layer is essential.

Enterprise networks typically consist of a number of remote sites connected to the headquarters campus via a WAN. Remote sites typically consist of a switched LAN, and the headquarters campus network is both routed and switched. Providing end-to-end QoS through such an environment requires that CoS markings that are set at the LAN edge be mapped into QoS markings (such as IP precedence or DSCP) for transit through campus or WAN routers. Campus and WAN routers can also map the QoS markings to new data-link headers for transit across the LAN. In this way, QoS can be preserved and uniformly applied across the enterprise.

Service providers offering IP services have a requirement to provide robust QoS solutions to their customers. The ability to map network-layer QoS to link-layer CoS allows these providers to offer a complete end-to-end QoS solution that does not depend on any specific link-layer technology.

Compatibility between an MPLS transport-layer and network-layer QoS is also achieved by mapping between MPLS EXP bits and the IP precedence or DSCP bits. A service provider can map the customer network-layer QoS marking as-is, or change it to fit an agreed-upon service level agreement (SLA). The information in the MPLS EXP bits can be carried end-to-end in the MPLS network, independent of the transport media. In addition, the network layer marking can

remain unchanged so that when the packet leaves the service provider MPLS network, the original QoS markings remain intact. Thus, a service provider with an MPLS network can help provide a true end-to-end QoS solution.

QoS Service Class Defined

This topic describes QoS service class and how service classes can be used to create a service policy throughout a network.

QoS Service Class Defined

- A QoS service class is a logical grouping of packets that are to receive a similar level of applied quality.
- A QoS service class can be a:
 - **Single user:** MAC address, IP address...
 - **Department, customer:** Subnet, interface...
 - **Application:** Port numbers, URL...

© 2003 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-10

When an administrative policy requiring QoS is created, you must determine how network traffic is to be treated. As part of that policy definition, network traffic must be associated with a specific service class. QoS classification mechanisms are used to separate traffic and identify packets as belonging to a specific service class. QoS marking mechanisms are used to tag each packet as belonging to the assigned service class. After the packets are identified as belonging to a specific service class, QoS mechanisms such as policing, shaping, and queuing techniques can be applied to each service class to meet the specifications of the administrative policy. Packets belonging to the same service class are given the same treatment with regards to QoS.

A QoS service class, being a logical grouping, can be defined in many ways, including these:

- Organization or department (marketing, engineering, sales, and so on)
- A specific customer or set of customers
- Specific applications or set of applications (Telnet, FTP, voice, Service Advertisement Protocol [SAP], Oracle, video, and so on)
- Specific users or sets of users (based on MAC address, IP address, LAN port, and so on)
- Specific network destinations (tunnel interfaces, virtual private networks [VPNs], and so on)

Example: Defining QoS Service Classes

A network administrator wants to apply QoS to the corporate network to better control bandwidth allocation of different network applications. Before QoS can be applied, an administrative QoS policy is first devised as follows:

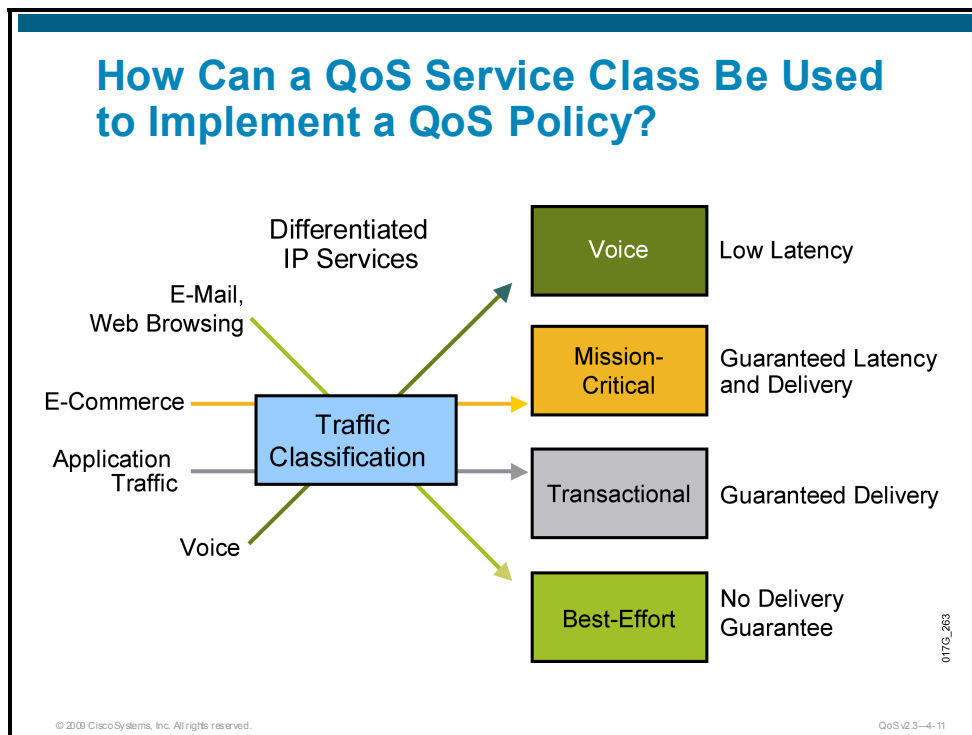
- Voice traffic is to be given a strict priority over all other traffic types.
- Business applications (FTP, TN3270, and Oracle) should be given priority over web traffic and have a guaranteed bandwidth of 20 percent.
- Web traffic should consume no more than 30 percent of any WAN link.

As a result of this policy, three QoS service classes have been defined:

- **Voice class:** To be treated with a strict priority service
- **Business applications class:** Requires a guaranteed bandwidth of 20 percent and is to be given priority over web traffic
- **Web class:** Only allowed to consume up to 30 percent of any WAN link

Implementing a QoS Policy Using a QoS Service Class

This topic describes how link-layer and network-layer markers are used to define QoS service classes and the different applications that can be represented by each of these service classes.



Specifying an administrative policy for QoS requires that a specific set of service classes be defined. QoS mechanisms are uniformly applied to these individual service classes to meet the requirements of the administrative policy. Because the application of QoS mechanisms is applied to different service classes and used to differentiate between applications, users, and traffic, the service class is a key component of a successful QoS implementation.

There are many different methods in which service classes can be used to implement an administrative policy. The first step is to identify the traffic that exists in the network and the QoS requirements for each traffic type. Then, traffic can be grouped into a set of service classes for differentiated QoS treatment in the network.

One popular model for the application of QoS service classes is the customer model which is typically used by service providers when referring to customer traffic. The customer model defines the following service classes (although many variations exist):

- **Voice service class:** Delivers low latency for voice services
- **Mission-critical service class:** Guarantees latency and delivery for the transport of mission-critical business applications such as Systems Network Architecture (SNA)
- **Transactional service class:** Guarantees delivery and is used for more general applications that are not as sensitive to delay, such as e-commerce
- **Best-effort service class:** Used to support small business, email, and other best-effort applications

Provisioning for Data: General Principles

- Profile applications to their basic network requirements.
- Do not over-engineer provisioning.
- Do not assign more than three applications to Mission-Critical or Transactional classes.
- Use proactive policies before reactive (policing) policies.
- Seek executive endorsement of relative ranking of application priority prior to rolling out QoS policies for data.

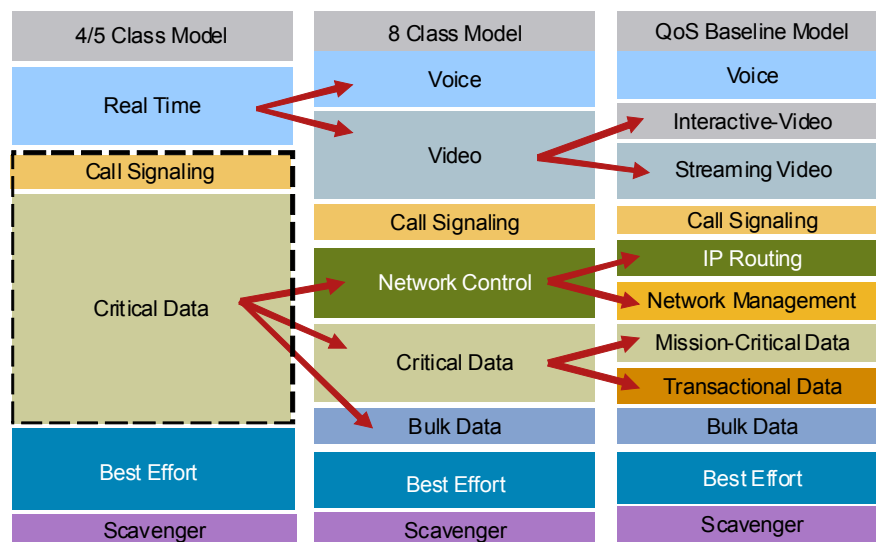
© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-12

One key element of defining QoS service classes is to understand the basic quality needs of network applications. It is essential that applications be given QoS treatment in line with their needs. For example, improperly specifying voice traffic into a service class with guaranteed bandwidth—without a guaranteed latency (delay)—would not meet the needs of the voice traffic.

While it is important to fully understand network application requirements, it is equally important not to over-provision or over-design the administrative policy. An administrative policy should be proactive in nature and require as few service classes as possible.

QoS Baseline Expansion



© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-13

The number of traffic classes used by enterprises has increased over the past few years, from 4 classes to 5–7 classes. The reason for this increase is that enterprises are using more and more applications and want increasingly more granularity in QoS differentiation among applications. The Cisco QoS baseline has suggested an 11-class model. This 11-class model is not mandatory, but merely an example of traffic classification based on various types of applications in use and their QoS requirements from an enterprise perspective. The Cisco AutoQoS for the Enterprise feature can automatically classify traffic into 10 of these 11 classes (the only class that the Cisco AutoQoS for the Enterprise feature does not automatically provision for is the “Mission Critical Data” class; this class is defined by subjective business requirements).

Example Application Service Classes

Application	L3 Classification			L2 CoS
	IPP	PHB	DSCP	
Routing	6	CS6	48	6
Voice	5	EF	46	5
Video Conferencing	4	AF41	34	4
Streaming Video	4	CS4	32	4
Mission-Critical Data	3	AF31*	26	3
Call Signaling	3	CS3*	24	3
Transactional Data	2	AF21	18	2
Network Management	2	CS2	16	2
Bulk Data	1	AF11	10	1
Scavenger	1	CS1	8	1
Best Effort	0	0	0	0

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-14

Although there are several sources of information that can be used as guidelines for determining a QoS policy, none of them can determine exactly what is proper for a specific network. Each network presents its own unique challenges and administrative policies. To properly implement QoS, measurable goals must be declared, and then a plan for achieving these goals must be formulated and implemented.

QoS must be implemented consistently across the entire network. It is not so important whether Call Signaling is marked as DSCP 34 or 26, but rather that DSCP 34 and 26 are treated in a manner that is necessary to accomplish the QoS policy. It is also important that data marked DSCP 34 is treated consistently across the network. If data travels over even a small portion of a network where different policies are applied (or no policies are applied), the entire QoS policy is nullified. Whether the data is crossing slow WAN links or Gigabit Ethernet, being switched by a Layer 2 switch or routed in a Layer 3 router, the policies must be consistently implemented to satisfy the policy requirements.

Originally, Cisco marked call signaling traffic as Assured Forwarding (AF) 31; call signaling traffic was originally marked by Cisco IP Telephony equipment to DSCP AF31. However, the Assured Forwarding classes, as defined in RFC 2597, were intended for flows that could be subject to markdown and, subsequently, the aggressive dropping of marked-down values. Marking down and aggressively dropping call signaling could result in noticeable delay-to-dial-tone (DDT) and lengthy call setup times, both of which generally translate to poor user experiences.

The Cisco QoS Baseline changed the marking recommendation for call signaling traffic to DSCP CS3 because Class Selector code points, as defined in RFC 2474, were not subject to markdown or aggressive dropping.

RFC 4594 Marking Recommendations

Cisco modified RFC 4594-based marking recommendations exchange Call Signaling and Broadcast Video, so that:

- Call Signaling = CS3 (DSCP 24)
- Broadcast Video = CS5 (DSCP 40)

Application	L3 Classification		IETF RFC
	PHB	DSCP	
Network Control	CS6	48	RFC 2474
VoIP Telephony	EF	46	RFC 3246
Call Signaling	CS5	40	RFC 2474
Multimedia Conferencing	AF41	34	RFC 2597
Real-Time Interactive	CS4	32	RFC 2474
Multimedia Streaming	AF31	26	RFC 2597
Broadcast Video	CS3	24	RFC 2474
Low-Latency Data	AF21	18	RFC 2597
OAM	CS2	16	RFC 2474
High-Throughput Data	AF11	10	RFC 2597
Best Effort	DF	0	RFC 2474
Low-Priority Data	CS1	8	RFC 3662

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-15

Since the development of the Cisco QoS Baseline, new RFCs have been published, such as RFC 4594, that improve and expand on the Cisco QoS Baseline. RFC 4594 was formally accepted in 2006 as an informational RFC. An informational RFC is an industry recommended best practice, while a standard RFC is an industry requirement. Therefore RFC 4594 is a set of formal DiffServ QoS configuration best practices, not a requisite standard.

RFC 4594 puts forward twelve application classes and matches them to RFC-defined per-hop behaviors (PHBs). These application classes and recommended PHBs are summarized in the figure. Obviously, Cisco QoS Baseline and RFC 4594 are similar, because RFC 4594 is essentially an industry-accepted evolution of the Cisco QoS Baseline. However, there are some differences that merit attention.

The first set of differences is minor, primarily involving nomenclature. RFC 4594 changes the names of some of the application classes from the QoS Baseline. These changes in nomenclature are summarized in the following table.

Nomenclature Changes from Cisco QoS Baseline to RFC 4594

Cisco QoS Baseline Class Names	RFC 4594 Class Names
Routing	Network Control
Voice	VoIP Telephony
Interactive Video	Multimedia Conferencing
Streaming Video	Multimedia Streaming
Transactional Data	Low-Latency Data
Network Management	Operation, Administration, and Management (OAM)
Bulk Data	High-Throughput Data
Scavenger	Low-Priority Data

The remaining changes, which follow, are more significant:

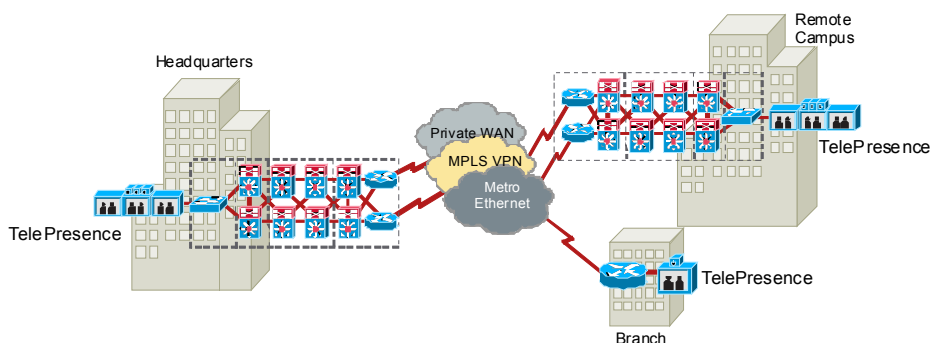
- The QoS Baseline Locally-Defined Mission-Critical Data class has been deleted from RFC 4594.
- The QoS Baseline marking recommendation of CS4 for Streaming Video has been changed in RFC 4594 to mark Multimedia Streaming to AF31.
- The QoS Baseline marking recommendation of CS3 for Call Signaling has been changed in RFC 4594 to mark Call Signaling to CS5.
- A new application class has been added to RFC 4594, Real-Time Interactive. This addition allows for a service differentiation between elastic conferencing applications (which would be assigned to the Multimedia Conferencing class) and inelastic conferencing applications (which would include high-definition applications, such as Cisco TelePresence, in the Real-Time Interactive class). Elasticity refers to the ability of the applications to function despite experiencing minor packet loss. Multimedia Conferencing uses the AF4 class and is subject to markdown (and potential dropping) policies, while the Real-Time Interactive class uses CS4 and is not subject to markdown or dropping policies.
- A second new application class has been added to RFC 4594, Broadcast Video. This addition allows for a service differentiation between elastic and inelastic streaming media applications. Multimedia Streaming uses the AF3 class and is subject to markdown (and potential dropping) policies, while Broadcast Video uses the CS3 class and is subject neither to markdown nor dropping policies.

The most significant of the differences between the Cisco QoS Baseline and RFC 4594 is the RFC 4594 recommendation to mark Call Signaling to CS5. Cisco has completed a lengthy marking migration for Call Signaling from AF31 to CS3 (as per the original QoS Baseline of 2002) and, as such, there are no plans to embark on another marking migration in the near future. It is important to remember that RFC 4594 is an informational RFC and not a standard. Therefore, Cisco plans to continue marking Call Signaling as CS3 until future business requirements arise that necessitate another marking migration.

Note	There are other RFCs relating to DiffServ design that are currently in draft status, such as RFC 5127, <i>Aggregation of Diffserv Service Classes</i> .

Classifying Cisco TelePresence

- A dedicated class, along with a dedicated classification marking value, is recommended for Cisco TelePresence.
- The recommended marking for Cisco TelePresence is CS4.



© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-10

Cisco TelePresence combines rich audio, high-definition video, and collaboration technologies to provide unique, real-time, face-to-face interactions between individuals in remote locations. A major benefit of Cisco TelePresence over competitive offerings is that the real-time, high-definition video and audio are transported over a converged IP network rather than a dedicated network (although dedicated networks are also supported). The key enabling technology to accomplish this convergence is QoS.

When implementing QoS for Cisco TelePresence, it is recommended that you designate a dedicated class along with a dedicated classification marking value for Cisco TelePresence. This is because Cisco TelePresence has higher service-level requirements than do generic videoconferencing and video telephony applications. The table below summarizes the service level requirements of both generic videoconferencing applications and Cisco TelePresence.

Service Level Requirements of Generic Video-Conferencing and Cisco TelePresence

Service Level Parameter (Target Values)	(Generic) Videoconferencing and Video Telephony	Cisco TelePresence
Bandwidth	384 kb/s or 768 kb/s plus network overhead	1.5 Mb/s to 12.6 Mb/s plus network overhead
Latency	400–450 ms latency	150 ms latency
Jitter	30–50 ms peak-to-peak jitter	10 ms peak-to-peak jitter
Loss	1% random packet loss	0.05% random packet loss

Videoconferencing and video telephony applications have traditionally been marked to RFC 2597 AF4, which is the recommendation from both the Cisco QoS Baseline as well as RFC 4594. However, the AF PHB includes policing, as well as increasing the Drop Preferences values, and ultimately dropping traffic according to the Drop Preference markings.

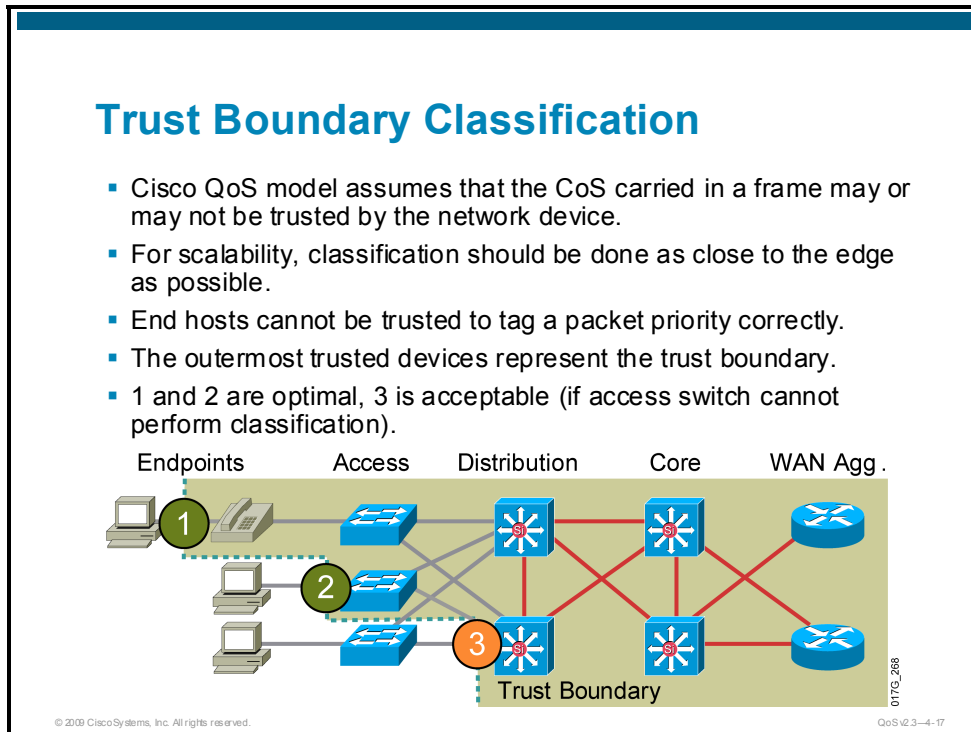
TelePresence traffic has a very low tolerance to drops (0.05%) and therefore would not be appropriately serviced by an AF PHB.

Because of the low-latency and jitter service-level requirements of Cisco TelePresence, it may seem attractive to assign it an RFC 3246 Expedited Forwarding (EF) PHB; there is nothing in RFC 3246 that dictates that only VoIP can be assigned to this PHB. However, it is important to recognize that VoIP behaves differently than video. VoIP has constant packet sizes and packet rates, whereas video packet sizes vary and video packet rates also vary in a random and bursty manner. If both video and voice were assigned to the same marking value and class, bursty video could easily interfere with well-behaved voice. Therefore, it is recommended in both the Cisco QoS Baseline and RFC 4594 that you avoid marking both voice and video to EF.

How, then, should Cisco TelePresence be marked? The best formal guidance is provided in RFC 4594, where a distinction is made between a Multimedia Conferencing (such as generic videoconferencing and video telephony) service class and a Real-Time Interactive service class. The Real-Time Interactive service class is intended for inelastic video flows, such as Cisco TelePresence. The recommended marking for this Real-Time Interactive service class, and thus the recommended marking for Cisco TelePresence, is CS4.

Trust Boundaries

This topic describes the concept of trust boundaries and how they are used with classification and marking.



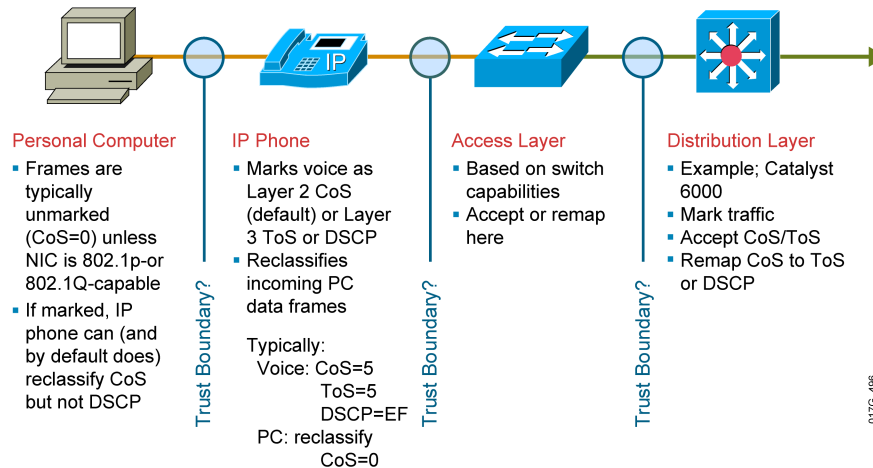
The concept of trust is important and integral to deploying QoS. After the end devices have set CoS or ToS values, the switch has the option of trusting them. If the switch trusts the values, it does not need to reclassify; if the switch does not trust the values, it must perform reclassification for the appropriate QoS.

The notion of trusting or not trusting forms the basis for the trust boundary. Ideally, classification should be done as close to the source as possible. If the end device is capable of performing this function, the trust boundary for the network is at the end device. If the device is not capable of performing this function, or the wiring closet switch does not trust the classification done by the end device, the trust boundary might shift.

How this shift happens depends on the capabilities of the switch in the wiring closet. If the switch can reclassify the packets, the trust boundary is in the wiring closet. If the switch cannot reclassify the packets, the task falls to other devices in the network, going toward the backbone. In this case, one good rule is to perform reclassification at the distribution layer, which means that the trust boundary shifts to the distribution layer. It is likely that there is a high-end switch in the distribution layer with features to support the reclassification function. If possible, try to avoid performing the reclassification function in the core of the network.

Trust Boundaries Mark Where?

- For scalability, marking should be done as close to the source as possible.



Classification should take place at the network edge, typically in the wiring closet or within endpoints (servers, hosts, video endpoints, or IP telephony devices).

For example, consider the campus network containing IP telephony and host endpoints. Frames can be marked as important by using link-layer CoS settings, or the IP precedence or DSCP bits in the ToS and DiffServ field in the IPv4 header. Cisco IP phones can mark voice packets as high priority using CoS as well as ToS. By default, the IP phone sends 802.1p-tagged packets with the CoS and ToS set to a value of 5 for its voice packets. Because most PCs do not have an 802.1Q-capable network interface card (NIC), they send packets untagged. This means that the frames do not have an 802.1p field. Also, unless the applications running on the PC send packets with a specific CoS value, this field is zero.

Note A special case exists in which the TCP/IP stack in the PC has been modified to send all packets with a ToS value other than zero. Typically, this does not happen, and the ToS value is zero.

Even if the PC is sending tagged frames with a specific CoS value, Cisco IP phones can zero out this value before sending the frames to the switch. This is the default behavior. Voice frames coming from the IP phone have a CoS of 5 and data frames coming from the PC have a CoS of 0. If the DSCP is set, then the IP phone cannot re-mark the DSCP.

If the end device is not a trusted device, the reclassification function (setting or zeroing the bits in the CoS and ToS fields) can be performed by the access layer switch, if that device is capable of doing so. If the device is not capable, then the reclassification task falls to the distribution layer device. If reclassification cannot be performed at one of these two layers (the access layer or the distribution layer), a hardware or Cisco IOS Software upgrade, or both, may be necessary.

Summary

This topic summarizes the key points discussed in this lesson.

Summary

- Packet classification is a QoS mechanism responsible for distinguishing between different traffic streams.
- Packet marking is a QoS mechanism that “colors” a packet so it can be distinguished from other packets during the application of QoS.
- Packets can be classified and marked at the data-link layer using several different mechanisms, including: 802.1Q, ISL, MPLS experimental bits, the Frame Relay DE bit, and the ATM CLP bit.
- Packets can be classified and marked at the network layer using IP precedence and DSCP.
- The ability to map network-layer QoS to link-layer CoS allows service providers to offer a complete end-to-end QoS solution that does not depend on any specific link-layer technology.

© 2003 Cisco Systems, Inc. All rights reserved.

QoS v2.3—4-19

Summary (Cont.)

- A QoS service class is a logical grouping of packets that, as specified in an administrative policy, are to receive a similar level of applied quality.
- An administrative policy for QoS requires that a specific set of service classes be defined. QoS mechanisms are uniformly applied to these individual service classes to meet the requirements of the administrative policy.
- It is important that a trust boundary be specified allowing classification and marking as close to the source as possible.

© 2003 Cisco Systems, Inc. All rights reserved.

QoS v2.3—4-20

Using MQC for Classification

Overview

The application of quality of service (QoS) requires that traffic be separated into service classes upon which you apply different levels of service. Separating traffic into different service classes requires QoS classification mechanisms. The Modular QoS CLI, or MQC, is one such mechanism for classifying network traffic. This lesson describes the packet classification features of the MQC, including input interface, access control lists (ACLs), class of service (CoS), IP precedence, and differentiated services code point (DSCP). The lesson also describes how you can configure MQC class maps to classify network traffic.

Objectives

Upon completing this lesson, you will be able to use MQC commands to classify packets. This ability includes being able to meet these objectives:

- Describe the different IP packet classification options in the MQC
- Identify the Cisco IOS commands used to configure classification of packets with MQC
- Identify the Cisco IOS commands used to monitor classification with MQC

MQC Classification Options

This topic describes the different IP packet classification options available in MQC.

MQC Classification Options

- Classification options configured in a class map
- Requires a referring policy map to be useful
- MQC classification options include the following:
 - Access list
 - IP precedence value
 - IP DSCP value
 - QoS group number
 - MPLS experimental bits
 - Protocol (including NBAR)
 - Using another class map
 - Frame Relay DE bit
 - IEEE 802.1Q/ISL CoS/Priority values
 - Input interface
 - Source MAC address
 - Destination MAC address
 - RTP (UDP) port range
 - Any packet

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.2

Classification using MQC is accomplished by specifying a traffic match criteria within a configured class map for each different service class. In order for QoS mechanisms to use the class map, the map must be referenced through the use of a policy map, which is subsequently applied to an inbound or outbound interface as a service policy.

In older Cisco IOS Software releases, the router classified a packet against every individual QoS feature. This resulted in additional processing overhead, inaccurate packet counters, and double accounting issues. Common classification is a feature that was introduced in Cisco IOS Software Release 12.2, and is enabled by default whenever classification is invoked within a policy map. With common classification, a packet is classified only once per service policy and matches a single class in the policy. Because matching terminates at the first matching class, it is important to ensure that the classes are configured in the right sequence within a policy. After a packet is classified against a particular class, that packet is subjected to all the QoS features configured within that class.

MQC classification with class maps is extremely flexible and can classify packets by using these classification tools:

- **ACLs:** ACLs for any protocol can be used within the class map configuration mode. The MQC can be used for other protocols, not only IP.
- **IP precedence:** IP packets can be classified directly by specifying IP precedence values.
- **DSCP:** IP packets can be classified directly by specifying IP DSCP values. DiffServ-enabled networks can have up to 64 classes if DSCP is used to mark packets.
- **MPLS experimental bits:** Packets can be matched based on the value in the experimental bits of the MPLS header of labeled packets.

- **QoS group:** A QoS group parameter can be used to classify packets in situations where up to 100 classes are needed or the QoS group parameter is used as an intermediate marker; for example, MPLS-to-QoS-group translation on input and QoS-group-to-DSCP translation on output. QoS group markings are local to a single router.
- **Protocol:** Classification is possible by identifying Layer 3 or Layer 4 protocols. Advanced classification is also available by using the Network-Based Application Recognition (NBAR) tool, which identifies dynamic protocols by inspecting higher-layer information.
- **Class map hierarchy:** Another class map can be used to implement template-based configurations.
- **Frame Relay DE bit:** Packets can be matched based on the value of the underlying Frame Relay DE bit.
- **CoS:** Packets can be matched based on the information contained in the three CoS bits (when using IEEE 802.1Q encapsulation) or priority bits (when using the Inter-Switch Link [ISL] encapsulation).
- **Input interface:** Packets can be classified based on the interface from which they enter the Cisco IOS device.
- **MAC address:** Packets can be matched based on their source or destination MAC addresses.
- **User Datagram Protocol (UDP) port range:** Real-Time Transport Protocol (RTP) packets can be matched based on a range of UDP port numbers.
- **All packets:** MQC can also be used to implement a QoS mechanism for all traffic in which case classification will put all packets into one class.

In Cisco IOS Software Release 12.4, packets can also be classified based on the following criteria:

- **Datagram header (Layer 2) or network header (Layer 3):** You can use the **match start** command to configure the match criteria for a class map on the basis of the datagram header (Layer 2) or the network header (Layer 3).
- **Discard Class:** You can use the **match discard-class** command to match packets of a certain discard class. A discard-class value has no mathematical significance. The value simply indicates that a packet marked with discard-class 2 should be treated differently than a packet marked with discard-class 1.
- **Field:** You can use the **match field** command to configure the match criteria for a class map on the basis of the fields defined in protocol header description files (PHDFs). Before configuring this match criterion, you must load a PHDF onto the router.
- **Frame Relay data-link connection identifier (DLCI):** You can use the **match fr-dlci** command to specify the Frame Relay DLCI number as a match criterion in a class map. This match criterion can be used in main interfaces and point-to-multipoint subinterfaces in Frame Relay networks. It can also be used in hierarchical policy maps.
- **MPLS experimental (EXP) value in the topmost label header:** You can use the **match mpls experimental topmost** command to match the MPLS EXP value in the topmost label. You can use this match criterion on the input interfaces and the output interfaces. It will match only on MPLS packets.
- **Packet length:** You can use the **match packet length** command to specify the Layer 3 packet length in the IP header as a match criterion in a class map.

- **Port type:** You can use the **match port-type** command to match traffic on the basis of the port type for a class map.
- **Tag:** You can use the **match tag** command to specify tag type as a match criterion.

Note As of Cisco IOS Software Release 12.2(31)SB2, you can classify traffic on the basis of a VLAN identification number on Cisco 10000 Series routers.

Configuring Classification with MQC

This topic describes the Cisco IOS commands used to configure classification of packets with MQC.

Configuring Classification with MQC

```
router(config)#
```

```
class-map [match-any | match-all] class-map-name
```

- Enters the class map configuration mode.
- Names can be a maximum of 40 alphanumeric characters.
- Match all is the default matching strategy.

```
router(config-cmap)#
```

```
match condition
```

- Use at least one condition to match packets.

```
router(config-cmap)#
```

```
match class-map class-map
```

- One class map can use another class map for classification.
- Nested class maps allow generic template class maps to be used in other class maps.

© 2009 Cisco Systems, Inc. All rights reserved.QoS v2.3-4.3

You can use the **class-map** global configuration command to create a class map and enter the class map configuration mode. A class map is identified by a case-sensitive name; therefore, all subsequent references to the class map must use exactly the same name.

You can use the **match** command to specify the classification criteria when in class map configuration mode. You can use multiple **match** commands within a class map. At least one **match** command should be used within the class map configuration mode. The default is **match none**.

You can also nest class maps in MQC configurations by using the **match class-map** command within the class map configuration. By nesting class maps, the creation of generic classification templates and more sophisticated classifications are possible.

The syntax for the **class-map** command is as follows:

```
class-map [match-any | match-all] class-map-name
```

Syntax Description

Parameter	Description
[match-any match-all]	(Optional) Determines how packets are evaluated when multiple match criteria exist. Packets must either meet all of the match criteria (match-all) or one of the match criteria (match-any) to be considered a member of the class. The default is " match-all ."
<i>class-map-name</i>	Name of the class for the class map. The name can be a maximum of 40 alphanumeric characters. The class name is used for both the class map and to configure policy for the class in the policy map.
Note	The command syntax shown in the figure is compatible with Cisco IOS Software Release 12.4.

Configuring Classification with MQC (Cont.)

```
router(config-cmap) #
```

```
match not match-criteria
```

- The “not” keyword inverts the condition.

```
router(config-cmap) #
```

```
match any
```

- The “any” keyword can be used to match all packets.

```
class-map Well-known-services
match access-group 100
!
Class-map Unknown-services
match not class-map Well-known-services
!
Class-map All-services
match any
!
access-list 100 permit tcp any any lt 1024
access-list 100 permit tcp any lt 1024 any
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.4

These are additional options that give extra power to class maps:

- Any condition can be negated by inserting the keyword **not**.
- A class map can use another class map to match packets.
- The **any** keyword can be used to match all packets.

The example in the figure shows these three class maps:

- Class map **Well-known-services** uses an ACL to match all the packets with the source or destination port number lower than 1024.
- Class map **Unknown-services** uses the first class map but negates the result. The same could be achieved by using the same ACL with a negation.
- Class map **All-services** actually matches all the packets.

The syntax for the **match not** command is as follows:

match not *match-criteria*

Syntax Description

Parameter	Description
<i>match-criteria</i>	(Required) Specifies the match-criterion value that is an unsuccessful match criterion. All other values of the specified match criteria will be considered successful match criteria.

Note The command syntax shown in the figure is compatible with Cisco IOS Software Release 12.4.

Configuring Classification Using Input Interface

```
router(config-cmap) #
```

```
match input-interface interface-name
```

- All packets received through the selected input interface are matched by this class map.

```
class-map match-any Ethernets
  match input-interface Ethernet0/0
  match input-interface Ethernet0/1
!
class-map match-any FastEthernets
  match input-interface FastEthernet1/0
  match input-interface FastEthernet1/1
!
class-map match-any Serials
  match input-interface Serial2/0
  match input-interface Serial2/1
  match input-interface Serial2/2
  match input-interface Serial2/3
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.6

As shown in the example in the figure, a packet can also be classified based on the input interface. In the first class-map example, called “Ethernets,” the **match input-interface** will match any packet that arrives on either the E0/0 or E0/1 interfaces.

In the second class map, “FastEthernets,” any packet arriving on either the FastEthernet 1/0 or FastEthernet 1/1 interface will be matched.

And in the last class map example, “Serials,” incoming packets arriving on any of S2/0, S2/1, S2/2, or S2/3 interfaces will be matched.

The syntax for the **match input-interface** command is as follows:

match input-interface *interface-name*

Syntax Description

Parameter	Description
<i>interface-name</i>	Name of the input interface to be used as match criteria

Note The syntax shown in the figure for the **match input-interface** command is the same as the syntax for the **match input-interface** command in Cisco IOS Software Release 12.4.

Configuring Classification Using CoS

```
router(config-cmap) #
```

```
match cos cos-value [cos-value cos-value cos-value]
```

- Select up to four CoS or priority values.
- Allowed values are 0 to 7.
- This classification option can only be used on interfaces using 802.1Q or ISL encapsulation.

```
class-map Strict-priority
  match cos 5
!
class-map High-priority
  match cos 4 6 7
!
class-map Low-priority
  match cos 0 1 2 3
```

© 2009 Cisco Systems, Inc. All rights reserved.

CoS v2.3-4.6

Routers can also match the three CoS bits in 802.1Q headers or priority bits in the ISL header. These bits can be used in a LAN-switched environment to provide differentiated quality of service.

This is demonstrated in the example. In the first class map, Strict-priority, packets will be matched if they have a CoS value of 5.

In the second class map example, High-priority, packets will be matched if they have a CoS value of 4, 6, or 7.

And in the last class map example, Low-priority, packets will be matched if they have a CoS value of 0, 1, 2, or 3.

match cos *cos-value* [*cos-value cos-value cos-value*]

Syntax Description

Parameter	Description
<i>cos-value</i>	(Optional) Specific IEEE 802.1Q/ISL CoS value. The <i>cos-value</i> is from 0 to 7; up to four CoS values can be specified in one match cos statement.

Note The command syntax shown in the figure is compatible with Cisco IOS Software Release 12.4.

Configuring Classification Using Access Lists

- **Access lists** are the oldest classification tool used with QoS mechanisms.
- Class maps support all types of access lists.
- Class maps are multiprotocol.
- Class maps can use named access lists and numbered access lists (in the range from 1 to 2699) for all protocols.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.7

ACLs were originally used for filtering of inbound or outbound packets on interfaces. They were later reused for filtering of routing updates and also for classification with early QoS tools, such as priority queuing (PQ), custom queuing (CQ), and traffic shaping.

ACLs are still one of the most powerful classification tools. Class maps can use any type of ACL (not only IP ACLs).

ACLs have a drawback: Compared to other classification tools, they are one of the most CPU-intensive. For this reason, ACLs should not be used for classification on high-speed links where they could severely impact performance of routers. ACLs are typically used on low-speed links at network edges, where packets are classified and marked (for example, with IP precedence). Classification in the core is done based on the IP precedence value.

However, ACLs may be efficiently used for classification purposes on Cisco Catalyst switches, because all classification operations (with the exception of NBAR classification on Cisco Catalyst 6500 Series) on Cisco Catalyst switches are performed in hardware, without any incremental CPU load.

Configuring Classification Using Access Lists (Cont.)

```
router(config-cmap) #
```

```
match access-group {number | name} [name]
```

- Select an access list to be used for classification.

```
class-map Telnet
  match access-group 100
!
class-map IPX_Printers
  match access-group name IPX_Printers
!
access-list 100 permit tcp any any eq 23
access-list 100 permit tcp any eq 23 any
!
ipx access-list sap IPX_Printers
  permit -1 7
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.8

Use the **match access-group** command to attach an ACL to a class map.

The example in the figure shows how a numbered or named ACL can be used for classification.

In the first example, class-map Telnet, packets will match according to those allowed by the access-group 100. When exploring what will be allowed, ACL 100 will permit port 23.

In the second example, packets will be allowed if they match according to those allowed by the ACL IPX_Printers.

match access-group {*number* | *name*} [*name*]

Syntax Description

Parameter	Description
<i>access-group</i>	A numbered ACL whose contents are used as the match criteria against which packets are checked to determine if they belong to this class
name <i>access-group-name</i>	A named ACL whose contents are used as the match criteria against which packets are checked to determine if they belong to this class

Configuring Classification Using IP Precedence

```
router(config-cmap) #
```

```
match ip precedence ip-prec-value [ip-prec [ip-prec [ip-prec]]]
```

- Select up to four IP precedence values or names.
- All packets marked with one of the selected IP precedence values are matched by this class map.

IP Precedence Value	IP Precedence Name
0	routine
1	priority
2	immediate
3	flash
4	flash-override
5	critical
6	Internet
7	network

0176_339

```
class-map VoIP
  match ip precedence 5
!
class-map Mission-Critical
  match ip precedence 3 4
!
class-map Transactional
  match ip precedence 1 2
!
class-map Best-Effort
  match ip precedence routine
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.9

A much faster method of classification is by matching the IP precedence. Up to four separate IP precedence values or names can be used to classify packets based on the IP Precedence field in the IP header on a single match-statement line.

The figure contains a mapping between IP precedence values and names. The running configuration, however, only shows IP precedence values (not names).

The syntax for the **match ip precedence** command is as follows:

match ip precedence *ip-prec-value* [*ip-prec* [*ip-prec* [*ip-prec*]]]

Note This command has been simplified to **match precedence** as of Cisco IOS Software Release 12.2[13]T. Inserting **ip** between **match** and **precedence** is optional as of this release and specifies that the match is only against IPv4 packets (if **ip** is omitted, the match will be against both IPv4 and IPv6 packets).

Syntax Description

Parameter	Description
<i>ip-precedence-value</i>	Specifies the exact value from 0 to 7 used to identify an IP precedence value.

Note The command syntax shown in the figure is compatible with Cisco IOS Software Release 12.4.

Configuring Classification Using DSCP

```
router(config-cmap) #
```

```
match [ip] dscp ip-dscp-value [ip-dscp-value ...]
```

- Select up to eight DSCP values or names.
- All packets marked with one of the selected DSCP values are matched by this class map.

DSCP Value	DSCP Class Name	DSCP Value	DSCP Class Name
0 (000000)	default	10 (001010)	af11
8 (001000)	cs1	12 (001100)	af12
16 (010000)	cs2	14 (001110)	af13
24 (011000)	cs3	18 (010010)	af21
32 (100000)	cs4	20 (010100)	af22
40 (101000)	cs5	22 (010110)	af23
48 (110000)	cs6	26 (011010)	af31
56 (111000)	cs7	28 (011100)	af32
46 (101110)	ef	30 (011110)	af33
		34 (100010)	af41
		36 (100100)	af42
		38 (100110)	af43

0170_487

© 2009 Cisco Systems, Inc. All rights reserved.

QoS V2.3-4-10

IP packets can also be classified based on the IP DSCP field. A QoS design can be based on IP precedence marking or DSCP marking. DSCP standards make IP precedence marking obsolete but include backward compatibility with IP precedence by using the Class Selector (CS) values (which are 6-bit equivalents to their IP precedence counterparts and are obtained by setting the three most significant bits of the DSCP to the IP precedence value, while holding the three least significant bits to zero).

The syntax for the **match [ip] dscp** command is as follows:

```
match [ip] dscp ip-dscp-value [ip-dscp-value ...]
```

Syntax Description

Parameter	Description
[ip]	(Optional) Specifies that the match is for IPv4 packets only. If not used, the match is on both IPv4 and IPv6 packets.
dscp-value	Specifies the exact value from 0 to 63 used to identify an IP DSCP value.

Note The command syntax shown in the figure is compatible with Cisco IOS Software Release 12.4.

Configuring Classification Using DSCP (Cont.)

```
class-map Voice
  match ip dscp ef cs5
!
class-map Mission-Critical
  match ip dscp af31 af32 af33 cs3
!
class-map Transactional
  match ip dscp af21 af22 af23 cs2
!
class-map Bulk
  match ip dscp af11 af12 af13 cs1
!
class-map Best-Effort
  match ip dscp default
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-11

The figure illustrates implementation of a design with these five classes:

- **Voice:** Identified by DSCP value EF, which looks like IP precedence value 5 in non-DSCP-compliant devices.
- **Mission-Critical:** Identified by DSCP values AF31, AF32, and AF33. The class is also identified by IP precedence 3.
- **Transactional:** Identified by DSCP values AF21, AF22, and AF23. The class is also identified by IP precedence 2.
- **Bulk:** Identified by DSCP values AF11, AF12, and AF13. The class is also identified by IP precedence 1.
- **Best-effort:** Identified by the default DSCP value that is equal to the default IP precedence value (0).

From a non-DSCP-compliant device, the design looks slightly different:

- **Voice:** IP precedence 5
- **Mission-Critical:** IP precedence 3
- **Transactional:** IP precedence 2
- **Bulk:** IP precedence 1
- **Best-effort:** IP precedence 0

Configuring Classification Using a UDP Port Range

```
router(config-cmap) #
```

```
match ip rtp starting-port-number port-range
```

- Use this command to implement classification equal to IP RTP Priority.
- All UDP packets with source or destination port numbers within the specified range are matched.
- Range is between the starting-port (values from 2000 to 65535) and the sum of the starting-port and the port-range (values from 0 to 16383).
- The command should be used in combination with class-based low-latency queuing to implement RTP Priority using MQC.

```
class-map RTP  
  match ip rtp 16384 16383
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS V2.3-4-12

IP RTP Priority was introduced to provide low latency queuing (LLQ) in combination with weighted fair queuing (WFQ). The **match ip rtp** command can be used to match packets in the same way as with IP RTP Priority. The command should also be combined with LLQ to generate a similar result as IP RTP Priority.

The syntax for the **match ip rtp** command is as follows:

match ip rtp *starting-port-number port-range*

Syntax Description

Parameter	Description
<i>starting-port-number</i>	The starting RTP port number. Values range from 2000 to 65535.
<i>port-range</i>	The RTP port number range. Values range from 0 to 16383.

Note The syntax shown in the figure for the **match ip rtp** command is the same as the syntax for the **match ip rtp** command in Cisco IOS Software Release 12.4.

Configuring IPv6 Classification via the MQC

- For IPv6, the MQC allows you to classify packets based on:
 - Precedence: Requires the **match precedence** command without the **ip** keyword; to match IPv6 packets only, also requires the **match protocol ipv6** command
 - DSCP: Requires the **match dscp** command without the **ip** keyword; to match IPv6 packets only, also requires the **match protocol ipv6** command
 - IPv6 access group values in IPv6 packets: Requires the **match access-group name** command (followed by the name of an IPv6 access group)

```
Router(config)# class-map IPP5
Router(config-cmap)# match protocol ipv6
Router(config-cmap)# match precedence 5
Router(config-cmap)# exit
Router(config)# policy-map PRIORITY50
Router(config-pmap)# class IPP5
Router(config-pmap-c)# priority 50
Router(config-pmap-c)# exit
Router(config-pmap)# exit
Router(config)# interface fa1/0/0
Router(config-if)# service-policy input PRIORITY50
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-13

To implement QoS in networks running IPv6, follow the same steps that you would follow to implement QoS in networks running only IPv4.

Classification in IPv6 can be based on IPv6 precedence, DSCP, and other IPv6 protocol-specific values that can be specified in IPv6 access lists. Classification can also be based on non-IPv6 protocol-specific values such as CoS, packet length, and QoS group. The MQC allows you to create matches on precedence, DSCP, and IPv6 access group values in IPv6 packets.

Except for the following special considerations, the functionality of all of the **match** commands is the same for IPv4 and IPv6:

- **Usage of match dscp command:** The **ip** keyword specifies that the match is for IPv4 packets only. If the **ip** keyword is not used, the match is on both IPv4 and IPv6 packets. To match DSCP values for IPv6 packets only, the **match protocol ipv6** command must also be used.
- **Usage of match precedence command:** The **ip** keyword specifies that the match is for IPv4 packets only. If the **ip** keyword is not used, the match is on both IPv4 and IPv6 packets. To match IP precedence values for IPv6 packets only, the **match protocol ipv6** command must also be used.
- **Usage of match access-group command:** To specify the name of an IPv6 access list against whose contents packets are checked to determine if they belong to the traffic class, use the **match access-group name** command in class-map configuration mode. The **name** keyword must be used and must be followed by the name of an IPv6 access group.

Note The Cisco 10000 Series Router supports DSCP matching of IPv4 packets only. You must include the **ip** keyword when specifying the DSCP values to use as match criterion.

Packet classification is available with both process and Cisco Express Forwarding switching paths. However, the **match cos** command for 802.1Q interfaces is supported only for Cisco Express Forwarding switched packets. Process-switched packets, such as router-generated packets, are not marked when this option is used.

Monitoring Class Maps

This topic describes the Cisco IOS commands that are used to monitor classification with MQC.

Monitoring Class Maps

```
router>  
show class-map [class-map-name]
```

- Displays all class maps and their matching criteria

```
router>show class-map  
Class Map match-any class-default (id 0)  
  Match any  
  
Class Map match-all Well-known-services (id 2)  
  Match access-group 100  
  
Class Map match-any All-services (id 4)  
  Match class-map Well-known-services  
  Match class-map Unknown-services  
  
Class Map match-all Unknown-services (id 3)  
  Match not class-map Well-known-services
```

© 2003 Cisco Systems, Inc. All rights reserved. QoS v2.3-4-14

The **show class-map** command lists all class maps with their match statements. This command can be issued from the EXEC or privileged EXEC mode. The **show class-map** command with a name of a class map displays the configuration of the selected class map.

In the figure, the **show class map** Cisco IOS command shows all the class maps that have been configured and what match statements are contained in the maps. The first class map listed is the default class map. The default class map contains only a single match statement, **match-any**. The second class map listed, Well-known-services, has one match statement that will compare packets against the configured access-group 100. The third class map displayed, All-services, contains two match statements that compare packets against two other configured class maps, Well-known-services and Unknown-services.

show class-map [class-map-name]

Syntax Description

Parameter	Description
class-map-name	(Optional) Name of the class map.

Note The command syntax shown in the figure is compatible with Cisco IOS Software Release 12.4.

Summary

This topic summarizes the key points discussed in this lesson.

Summary

- MQC classification options include: access list, IP precedence value, IP DSCP value, QoS group number, MPLS experimental bits, protocol (including NBAR), using another class map, Frame Relay DE bit, IEEE 802.1Q/ISL CoS/Priority values, input interface, source MAC address, destination MAC address, RTP (UDP) port range, and any packet.
- The MQC uses class maps to specify match criteria, allowing classification of traffic for QoS treatment.
- Class maps are monitored by using the **show class-map** command.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-15

Using MQC for Class-Based Marking

Overview

The process of packet classification is both complex and CPU-intensive. You should classify packets as close to the source as possible—at the edges of the network. Performing classification in the core is not recommended because it would add delay in transiting the core. To provide different levels of treatment to service classes, you must identify traffic as “belonging” to a specific class. Instead of classifying traffic at each hop in the network as the packet traverses the network to its ultimate destination, QoS marking mechanisms are used. Marking allows you to set specific fields in a frame or packet that identify that frame or packet as belonging to a specific service class. The Modular quality of service (QoS) command-line interface (CLI), or MQC, provides one such mechanism for marking network traffic. This lesson describes the class-based marking capability of the Cisco IOS MQC and how policy maps can be configured to mark network traffic. MQC marking features covered in this lesson include class of service (CoS), IP precedence, and differentiated services code point (DSCP).

Objectives

Upon completing this lesson, you will be able to use class-based marking to assign packets to a specific service class. This ability includes being able to meet these objectives:

- Describe the MQC class-based marking system
- Describe the different IP packet-marking options available in class-based marking
- Identify the Cisco IOS commands required to configure class-based marking
- Identify the Cisco IOS commands used to monitor class-based marking

Class-Based Marking Overview

This topic describes the MQC class-based marking mechanism.

Class-Based Marking Overview

- **Class-based marking** is an additional tool available with the MQC that allows static per-class marking of packets.
- It can be used to mark inbound or outbound packets.
- It can be combined with any other QoS feature on output.
- It can be combined with class-based policing on input.
- Cisco Express Forwarding must be configured on the interface before the class-based packet marking feature can be used.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.2

Marking packets or frames sets information in the Layer 2 and Layer 3 headers of a packet so that the packet or frame can be identified and distinguished from other packets or frames.

MQC provides packet-marking capabilities using class-based marking. MQC is the most flexible Cisco IOS marking tool, extending the marking functionality of committed access rate (CAR) and policy routing.

You can use class-based marking on input or output of interfaces as part of a defined input or output service policy. On input, you can combine class-based marking with class-based policing, and on output, with any other class-based weighted fair queuing (CBWFQ) QoS feature.

MQC Marking Options

This topic describes the different IP packet and frame marking options that are available in class-based marking.

MQC Marking Options

Packets can be marked with one of these:

- IP precedence
- IP DSCP
- QoS group
- MPLS experimental bits
- IEEE 802.1Q or ISL CoS/priority bits
- Frame Relay DE bit
- ATM CLP bit

© 2009 Cisco Systems, Inc. All rights reserved.QoS v2.3-4.3

Class-based marking supports these markers:

- IP precedence
- IP DSCP value
- QoS group
- Multiprotocol Label Switching (MPLS) experimental bits
- IEEE 802.1Q or Inter-Switch Link (ISL) CoS or priority bits
- Frame Relay discard eligible (DE) bit
- ATM cell loss priority (CLP) bit

Note

A QoS group is an internal label used by a switch to identify packets as members of a specific class. The label is not part of the packet header and is restricted to the switch that sets the label. QoS groups provide a way to tag a packet for subsequent QoS action without explicitly marking (changing) the packet. You can then communicate an ACL match from an input policy map to an output policy map.

In Cisco IOS Software Release 12.4, class-based marking also supports these markers:

- Discard-class value
- MPLS EXP field value in the topmost label on either an input or an output interface
- ToS bits in the header of an IP packet

You can combine class-based marking with other mechanisms available in the MQC.

Configuring Class-Based Marking

This topic describes the Cisco IOS commands that are required to configure class-based marking.

Configuring Class-Based Marking

```
router(config)#
```

```
class-map {match-any | match-all} class-map-name
```

1. **Create class map:** A traffic class (match access list, input interface, IP precedence, DSCP, protocol [NBAR] src/dst MAC address).

```
router(config)#
```

```
policy-map policy-map-name
```

2. **Create policy map (service policy):** Associate a class map with one or more QoS marking policies.

```
router(config-if)#
```

```
service-policy {input | output} policy-map-name
```

3. **Attach service policy:** Associate the policy map to an input or output interface.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.4

When configuring class-based marking, you must complete these three configuration steps:

- Step 1** Create a class map.
- Step 2** Create a policy map.
- Step 3** Attach the policy map to an interface by using the **service-policy** Cisco IOS command.

The syntax for the **class-map** command is as follows:

```
class-map {match any | match all} class-map-name
```

Syntax Description

Parameter	Description
<i>class-map-name</i>	Name of the class for the class map. The class name is used for both the class map and to configure policy for the class in the policy map.
match-any match-all	Determines how packets are evaluated when multiple match criteria exist. Packets must either meet all of the match criteria (match-all) or one of the match criteria (match-any) to be considered a member of the class.

The syntax for the **policy-map** command is as follows:

```
policy-map policy-map-name
```

Syntax Description

Parameter	Description
<i>policy-map-name</i>	Name of the policy map

The syntax for the **service-policy** command is as follows:

```
service-policy {input | output} policy-map-name
```

Syntax Description

Parameter	Description
<i>policy-map-name</i>	Specifies the name of the predefined policy map to be used as a QoS policy. The name can be a maximum of 40 alphanumeric characters.

Note	The command syntax shown in the figure is compatible with Cisco IOS Software Release 12.4.

Configuring Class-Based Marking (Cont.)

```
class-map Well-known-services
  match access-group 100
!
class-map Unknown-services
  match not class-map Well-known-services
!
policy-map set-DSCP
  class Well-known-services
    set DSCP AF21
  class Unknown-services
    set DSCP 0
!
access-list 100 permit tcp any any lt 1024
access-list 100 permit tcp any lt 1024 any
!
Interface ethernet 0/0
  service-policy input set-DSCP
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.5

In the figure, two class maps have been configured, Well-known-services and Unknown-services. The match criteria is specified in access-list 100.

The policy map, set-DSCP, has been created to associate with the class maps, Well-known-services and Unknown-services.

For packets allowed by class map Well-known-services, the IP DSCP value will be set to AF21. The matching class map, Unknown-services, will have the IP DSCP value set to 0.

The policy map is attached to E0/0 for incoming packets by the **service-policy** command.

Configuring CoS Marking

```
router(config-pmap-c) #
```

```
set cos cos-value
```

- Mark frames with the specified value (0 to 7).
- The value applies to the CoS bits with the IEEE 802.1Q encapsulation or priority bits with the ISL encapsulation.
- The command can only be used on LAN interfaces that use one of the two mentioned encapsulations.

```
policy-map SetCoS
class Class1
  set cos 1
class Class2
  set cos 2
class Class3
  set cos 3
```

© 2009 Cisco Systems, Inc. All rights reserved.

CoS v2.3-4.6

The IEEE 802.1p specification specifies a standard for delivering QoS in LANs. Packets are marked with three CoS bits, with values ranging from zero for low priority to seven for high priority. CoS can only be applied on trunks because VLAN trunking encapsulations designate fields with available space to carry CoS bits. There are currently two widely deployed trunking protocols that can transport CoS markings, as follows:

- **ISL frame headers:** ISL frame headers have a 1-byte user field that carries the CoS value in the three least significant bits.
- **IEEE 802.1p and 802.1Q frame headers:** IEEE 802.1p and 802.1Q frame headers have a 2-byte Tag Control Information (TCI) field that carries the CoS value in the three most significant bits, called the user priority bits.

Note Other (untagged) Ethernet frame types cannot carry CoS values.

In general, legacy Layer 2 switches can examine, use, or alter MAC layer markings (but not IP precedence or DSCP settings), because IP precedence and DSCP are Layer 3. Layer 2 markings are generally applied on egress trunk ports.

The syntax for the **set cos** command is as follows:

set cos *cos-value*

Syntax Description

Parameter	Description
<i>cos-value</i>	Specific IEEE 802.1Q CoS value from 0 to 7

The syntax for the **set cos** command in Cisco IOS Software Release 12.4 is as follows:

```
set cos {cos-value | from-field [table table-map-name]}
```

Syntax Description

Parameter	Description
<i>cos-value</i>	Specific IEEE 802.1Q CoS value from 0 to 7.
<i>from-field</i>	Specific packet-marking category to be used to set the CoS value of the packet. If you are using a table map for mapping and converting packet-marking values, this establishes the "map from" packet-marking category. Packet-marking category keywords are as follows: ■ precedence ■ dscp
table	(Optional) Indicates that the values set in a specified table map will be used to set the CoS value.
<i>table-map-name</i>	(Optional) Name of the table map used to specify the CoS value. The table map name can be a maximum of 64 alphanumeric characters.

Note	In addition to using individual set commands for specifying and marking traffic attributes, you can specify and mark the traffic attribute by creating a mapping table (called a "table map"). With this method, you configure the traffic attributes that you want to mark once in a table map and then the markings can be propagated throughout the network. For more information on the table map method, refer to <i>Cisco IOS Quality of Service Solutions Configuration Guide</i> , Release 12.4, at http://www.cisco.com .

Configuring IP Precedence Marking

```
router(config-pmap-c) #
```

```
set ip precedence ip-precedence-value
```

- Mark IP packets with the specified IP precedence value.
- IP precedence can be set using a value (0 to 7) or a corresponding name (for example, routine, priority, immediate).

```
policy-map SetPrec
class Class1
  set ip precedence priority
class Class2
  set ip precedence flash
class Class3
  set ip precedence 5
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.7

IP precedence is encoded into the three high-order bits of the type of service (ToS) field in the IP header. It supports eight classes, two of which (IP precedence 6 and 7) are reserved for control-plane traffic and should not be used for user-defined classes. IP precedence 0 is the default value and is usually used for the best-effort class.

To set the precedence value in the IP header, use the **set ip precedence** QoS policy-map configuration command. To leave the precedence value at the current setting, use the **no** form of this command.

The syntax for the **set ip precedence** command is as follows:

set ip precedence *ip-precedence-value*

Syntax Description

Parameter	Description
<i>ip-precedence-value</i>	A number from 0 to 7 that sets the precedence bit in the IP header

In Cisco IOS Software Release 12.4, use the **set precedence** command to set the precedence value in the packet header. The syntax for the **set precedence** command is as follows:

```
set precedence {precedence-value | from-field [table table-map-name]}
```

Syntax Description

Parameter	Description
<i>precedence-value</i>	A number from 0 to 7 that sets the precedence bit in the packet header.
<i>from-field</i>	Specific packet-marking category to be used to set the precedence value of the packet. If you are using a table map for mapping and converting packet-marking values, this argument value establishes the "map from" packet-marking category. Packet-marking category keywords are as follows: ■ cos ■ qos-group
table	(Optional) Indicates that the values set in a specified table map will be used to set the precedence value.
<i>table-map-name</i>	(Optional) Name of the table map used to specify a precedence value based on the CoS value. The name can be a maximum of 64 alphanumeric characters.

To set the precedence values for IPv4 packets only, include the **match protocol ip** command along with the other commands in the class map. Without the **match protocol ip** command, the class map may match both IPv6 and IPv4 packets (depending on the other match criteria) and the **set precedence** command may act upon both types of packets.

Configuring IP DSCP Marking

```
router(config-pmap-c) #
```

```
set [ip] dscp ip-dscp-value
```

- Mark IP packets with the specified DSCP value.
- DSCP can be set using a value (0 to 63) or a corresponding name (for example, af11, af12, af13, af21, ef, cs1, default).

```
policy-map SetDSCP
class Class1
  set ip dscp af11
class Class2
  set ip dscp af21
class Class3
  set ip dscp ef
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.8

Differentiated Services (DiffServ) is a standards-based Layer 3 marking model that supersedes—and is backward compatible with—IP precedence. DiffServ uses six prioritization bits that permit classification of up to 64 values (0 to 63). A DiffServ value is called a DSCP. The **set ip dscp** command is used to mark packets of a class with a DSCP value.

To mark a packet by setting the IP DSCP in the ToS byte, use the **set ip dscp** QoS policy-map configuration command. To remove a previously set IP DSCP, use the **no** form of this command.

The syntax for the **set [ip] dscp** command is as follows:

```
set [ip] dscp ip-dscp-value
```

Syntax Description

Parameter	Description
<i>ip-dscp-value</i>	A number from 0 to 63 that sets the IP DSCP value. Reserved keywords EF (Expedited Forwarding), AF11 (Assured Forwarding Class 1 Drop Preference 1), AF12 (Assured Forwarding Class 1 Drop Preference 2), and so on can be specified instead of numeric values.

The syntax for this command in Cisco IOS Software Release 12.4 is as follows:

```
set [ip] dscp {dscp-value | from-field [table table-map-name]}
```


Syntax Description

Parameter	Description
ip	(Optional) Specifies that the match is for IPv4 packets only. If not used, the match is on both IPv4 and IPv6 packets.
<i>ip-dscp-value</i>	A number from 0 to 63 that sets the DSCP value. The following keywords are examples of reserved keywords that can be specified instead of numeric values: ■ EF (expedited forwarding) ■ AF11 (assured forwarding class AF11) ■ AF12 (assured forwarding class AF12)
<i>from-field</i>	Specific packet-marking category to be used to set the DSCP value of the packet. If you are using a table map for mapping and converting packet-marking values, this establishes the "map from" packet-marking category. Packet-marking category keywords are as follows: ■ cos ■ qos-group
table	(Optional) Used in conjunction with the <i>from-field</i> argument. Indicates that the values set in a specified table map will be used to set the DSCP value.
<i>table-map-name</i>	(Optional) Used in conjunction with the table keyword. Name of the table map used to specify the DSCP value. The name can be a maximum of 64 alphanumeric characters.

Class-Based Packet Marking in IPv6 Networks

- You can create policies to mark packets by **DSCP** or **precedence**.
- Use the **set dscp** or **set precedence** command in a policy map to configure packet marking.
 - Use the **set dscp** command without the **ip** keyword.
 - To set DSCP or precedence values for IPv6 packets only, use the **match protocol ipv6** command in the class-map that classifies packets for this action.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.9

For IPv6 networks, you can create policies to mark packets by DSCP or precedence. Use the **set dscp** or **set precedence** command in a policy map to configure packet marking.

To use the **set dscp** command for packet marking in an IPv6 network, use the command without the **ip** keyword. Use of the **ip** keyword specifies that the match is for IPv4 packets only. To set the DSCP values for IPv6 packets only, the **match protocol ipv6** command must also be used in the class map that classified packets for this action.

When the **set precedence** command is used in IPv6 environments, it can set the value in both IPv4 and IPv6 packets. However, the actual packets set by this function are only those that meet the match criteria of the class map containing this function. To set the precedence values for IPv6 packets only, the **match protocol ipv6** command must also be used in the class map that classified packets for this action. Without the **match protocol ipv6** command, the class map may classify both IPv6 and IPv4 packets, (depending on other match criteria) and the **set precedence** command will act upon both types of packets.

Note Cisco IOS **set** commands can vary by release. For more information, see the command documentation for the Cisco IOS Software release that you are using.

Monitoring Class-Based Marking

This topic describes the Cisco IOS commands that are used to monitor class-based marking.

Monitoring Class-Based Marking

```
Router>  
show policy-map [policy-map]
```

- Displays the configuration of all classes for a specified service policy map or all classes for all existing policy maps.

```
router# show policy-map  
  
Policy Map SetCoS  
Class Class1  
  set cos 1  
Class Class2  
  set cos 2  
Class Class3  
  set cos 3  
Class Class4  
  set cos 4  
Class Class5  
  set cos 5  
Class Class6  
  set cos 6  
Class Class7  
  set cos 7
```

© 2009 Cisco Systems, Inc. All rights reserved. QoS v2.3-4.10

The **show policy-map** command displays all classes for the service policy specified in the command line.

To display the configuration of all classes for a specified service policy map or all classes for all existing policy maps, use the **show policy-map** EXEC or privileged EXEC command.

The syntax for the **show policy-map** command is as follows:

```
show policy-map [policy-map]
```

Syntax Description

Parameter	Description
<i>policy-map</i>	(Optional) The name of the service policy map whose complete configuration is to be displayed

The syntax shown in the figure for the **show policy-map** command is the same as the syntax for the **show policy-map** command in Cisco IOS Software Release 12.4.

Monitoring Class-Based Marking (Cont.)

Router>

```
show policy-map interface interface-name
```

- Displays the configuration of all classes configured for all service policies on the specified interface.

```
router# show policy-map interface serial 0/0
Serial0/0

Service-policy input: SetMPLS (1837)

Class-map: Class1 (match-any) (1839/12)
  0 packets, 0 bytes
  30 second offered rate 0 bps, drop rate 0 bps
Match: qos-group 1 (1843)
  0 packets, 0 bytes
  30 second rate 0 bps
QoS Set
    mpls experimental 1
...
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-11

The **show policy-map interface** command displays all service policies applied to the interface. In addition to the settings, marking parameters and statistics are displayed.

To display the configuration of all classes configured for all service policies on the specified interface or to display the classes for the service policy for a specific permanent virtual circuit (PVC) on the interface, use the **show policy-map interface EXEC** or privileged EXEC command.

The syntax for the **show policy-map interface** command is as follows:

```
show policy-map interface interface-name [vc [vpi/] vci] [dlci
dlci] [input | output]
```

Syntax Description

Parameter	Description
<i>interface-name</i>	Name of the interface or subinterface whose policy configuration is to be displayed.
<i>vpi</i> /	(Optional) ATM network virtual path identifier (VPI) for this PVC. The absence of the "/" and a <i>vpi</i> value defaults the <i>vpi</i> value to 0. On the Cisco 7200 and 7500 Series Routers, this value ranges from 0 to 255. The <i>vpi</i> and <i>vci</i> arguments cannot both be set to 0; if one is 0, the other cannot be 0. If this value is omitted, information for all virtual circuits (VCs) on the specified ATM interface or subinterface is displayed.

Parameter	Description
<i>vci</i>	(Optional) ATM network virtual channel identifier (VCI) for this PVC. This value ranges from 0 to one less than the maximum value set for this interface by the atm vc-per-vp command. Typically, lower values 0 to 31 are reserved for specific traffic (Operation, Administration, and Maintenance [OAM]; switched virtual circuit [SVC] signaling, Integrated Local Management Interface [ILMI], and so on) and should not be used. The VCI is a 16-bit field in the header of the ATM cell. The VCI value is unique only on a single link, not throughout the ATM network, because it has local significance only. The <i>vpi</i> and <i>vci</i> arguments cannot both be set to 0; if one is 0, the other cannot be 0.
dlci	(Optional) Indicates a specific PVC for which policy configuration will be displayed.
<i>dlci</i>	(Optional) A specific data-link connection identifier (DLCI) number used on the interface. Policy configuration for the corresponding PVC will be displayed when a DLCI is specified.
input	(Optional) Indicates that the statistics for the attached input policy will be displayed.
output	(Optional) Indicates that the statistics for the attached output policy will be displayed.

The syntax shown in the figure for the **show policy-map interface** command is compatible with the syntax for the **show policy-map interface** command in Cisco IOS Software Release 12.4.

Note	Syntax for commands documented in this lesson may vary by platform and software version. For more information on command syntax for your Cisco IOS Software release and router platform, refer to the <i>Cisco IOS Master Command List</i>, Release 12.4, at http://www.cisco.com.

Summary

This topic summarizes the key points discussed in this lesson.

Summary

- Class-based marking, which requires that Cisco Express Forwarding be enabled, can mark inbound or outbound packets.
- IP packet-marking options available in class-based marking include IP precedence, IP DSCP, QoS group, Layer 2 CoS value, and MPLS experimental bits.
- Class-based marking requires three configuration steps: class map, policy map, service policy; use the **set cos**, **set ip precedence**, and **set ip dscp** commands to set (mark) Layer 2 CoS, precedence, and DSCP values, respectively.
- Use the **show policy-map** and **show policy-map interface** commands to monitor class-based marking.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-12

Using NBAR for Classification

Overview

Network-Based Application Recognition (NBAR), a feature in Cisco IOS Software, provides intelligent classification for the network infrastructure. NBAR is a classification engine that can recognize a wide variety of protocols and applications, including web-based applications and client and server applications that dynamically assign TCP or User Datagram Protocol (UDP) port numbers. After the protocol or application is recognized, the network can invoke specific services for that particular protocol or application. NBAR currently works with quality of service (QoS) features to ensure that the network bandwidth is best used to fulfill company objectives. These features include the ability to guarantee bandwidth to critical applications, limit bandwidth to other applications, drop selected packets to avoid congestion, and mark packets appropriately so that the network and the service provider network can provide QoS from end to end.

This lesson describes NBAR, including applications that NBAR can support, Packet Description Language Modules (PDLMs), and NBAR Protocol Discovery.

Objectives

Upon completing this lesson, you will be able to use NBAR to discover network protocols and to classify packets. This ability includes being able to meet these objectives:

- Describe the Cisco IOS protocol discovery and classification mechanism known as NBAR
- Identify the types of applications supported by NBAR
- Explain the purpose of PDLMs in NBAR
- Explain how custom protocols can be used to extend the set of protocols recognized by NBAR
- Describe NBAR Protocol Discovery and the Cisco NBAR Protocol Discovery MIB
- Identify the Cisco IOS commands required to configure and monitor NBAR

Network-Based Application Recognition

This topic describes NBAR, a Cisco IOS protocol discovery and classification mechanism.

Network-Based Application Recognition

- NBAR solves the problem of how to classify modern client/server and web-based applications.
- NBAR performs the following functions:
 - Identification of applications and protocols (Layer 4 to Layer 7)
 - Protocol discovery
 - Provides traffic statistics
- NBAR enables downstream actions based on QoS policies via class-based queuing, policing, and shaping by selecting traffic which can then be marked to trigger downstream per-hop behaviors.
- New applications are easily supported by loading PDLM.

© 2009 Cisco Systems, Inc. All rights reserved.QoS v2.3-4.2

NBAR is a classification engine that recognizes and classifies a wide variety of protocols and applications, including web-based and other difficult-to-classify applications and protocols that use dynamic TCP/UDP port assignments.

When NBAR recognizes and classifies a protocol or application, the network can be configured to apply the appropriate QoS for that application or traffic with that protocol. The QoS is applied using the Modular QoS CLI, or MQC.

Examples of the QoS features that can be applied to the network traffic (using the MQC), after NBAR has recognized and classified the application or protocol, include the following:

- Class-based marking
- Class-based weighted fair queuing (CBWFQ)
- Low-latency queuing (LLQ)
- Traffic policing
- Traffic shaping

For Cisco IOS Software Release 12.2(18)ZYA on the Cisco Catalyst 6500 Series Switch (which is equipped with a Supervisor 32/Programmable Intelligent Services Accelerator [PISA]), only the QoS features listed below can be configured. These features can be configured (using the MQC) after NBAR has recognized and classified the application or protocol.

- Traffic classification
- Traffic marking

■ Traffic policing

NBAR includes a Protocol Discovery feature that provides an easy way to discover application protocols that are operating on an interface. The Protocol Discovery feature discovers any protocol traffic supported by NBAR. You can apply Protocol Discovery to interfaces and use it to monitor both input and output traffic. Protocol Discovery maintains per-protocol statistics for enabled interfaces such as total number of input and output packets and bytes, and input and output bit rates.

You can load an external PDLM at run time to extend the NBAR list of recognized protocols. PDLMs allow NBAR to recognize new protocols without requiring a new Cisco IOS image or a router reload.

NBAR introduces powerful application classification features into the network at a small-to-medium CPU overhead cost. The CPU utilization will vary based on factors such as the router processor speed and type, and the traffic rate.

Note	NBAR classifies network traffic by application or protocol. Network traffic can be classified without using NBAR.

NBAR gives you the ability to see the variety of protocols and the amount of traffic generated by each protocol. After gathering this information, NBAR allows you to organize traffic into classes.

NBAR Requirements and Restrictions

- Before you configure NBAR, you must enable Cisco Express Forwarding.
- NBAR is not supported on the following logical interfaces:
 - Fast EtherChannel
 - Dialer interfaces before Cisco IOS Release 12.2(4)T
 - Interfaces where tunneling or encryption is used
- NBAR does not support the following:
 - More than 24 concurrent URLs, hosts, or MIME-type matches
 - Non-IP traffic
 - Multicast and switching modes other than Cisco Express Forwarding
 - Fragmented packets
 - URL, host, or MIME classification with secure HTTP
 - Packets originating from or destined to the router running NBAR

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.3

The following requirements and restrictions apply to NBAR:

- Before you configure NBAR, you must enable Cisco Express Forwarding. This prerequisite does not apply if you are using Cisco IOS Software Release 12.2(18)ZYA.
- NBAR does not support the following:
 - More than 24 concurrent URLs, hosts, or Multipurpose Internet Mail Extension (MIME)-type matches. For Cisco IOS Software Release 12.2(18)ZYA, the maximum number of concurrent URLs, hosts, or MIME-type matches is 56.
 - Matching beyond the first 400 bytes in a packet payload in Cisco IOS Software releases before Cisco IOS Software Release 12.3(7)T. In Cisco IOS Software Release 12.3(7)T, this restriction was removed, and NBAR now supports full payload inspection. The only exception is that NBAR can inspect custom protocol traffic for only 255 bytes into the payload.
 - Non-IP traffic.
 - Multiprotocol Label Switching (MPLS)-labeled packets. NBAR classifies IP packets only. You can, however, use NBAR to classify IP traffic before the traffic is handed over to MPLS.
 - Multicast and switching modes other than Cisco Express Forwarding.
 - Fragmented packets.
 - Pipelined persistent HTTP requests.
 - URL, host, or MIME classification with secure HTTP.
 - Asymmetric flows with stateful protocols.
 - Packets that originate from or that are destined to the router running NBAR.
- NBAR is not supported on the following logical interfaces:
 - Fast EtherChannel

Note Fast EtherChannels are supported in Cisco IOS Software Release 12.2(18)ZYA.

- Dialer interfaces before Cisco IOS Software Release 12.2(4)T
- Interfaces where tunneling or encryption is used

Note You cannot use NBAR to classify output traffic on a WAN link where tunneling or encryption is used. Therefore, you should configure NBAR on other interfaces on the router (such as a LAN link) to perform input classification before the traffic is switched to the WAN link for output. However, NBAR protocol discovery is supported on interfaces on which tunneling or encryption is used. You can enable protocol discovery directly on the tunnel or on the interface on which encryption is performed to gather key statistics about the various applications that are traversing the interface. The input statistics also show the total number of encrypted or tunneled packets received in addition to the per-protocol breakdowns.

- NBAR is currently not supported with Stateful Switchover (SSO). This restriction applies to the Cisco Catalyst 6500 Series Switches and to the Cisco 7500 and Cisco 7600 Series Routers.
- To use Distributed NBAR (dNBAR) on a Cisco 7500 Series Router, you must use a slot controller (or VIP processor) that has 64 MB of DRAM or more.

Note When NBAR is used on a Cisco 7500 Series Router with a Versatile Interface Processor (VIP) or on a Catalyst 6500 Series Switch with a FlexWAN module or serial interface processor (SIP), NBAR is referred to as dNBAR. The implementation of NBAR and dNBAR is identical.

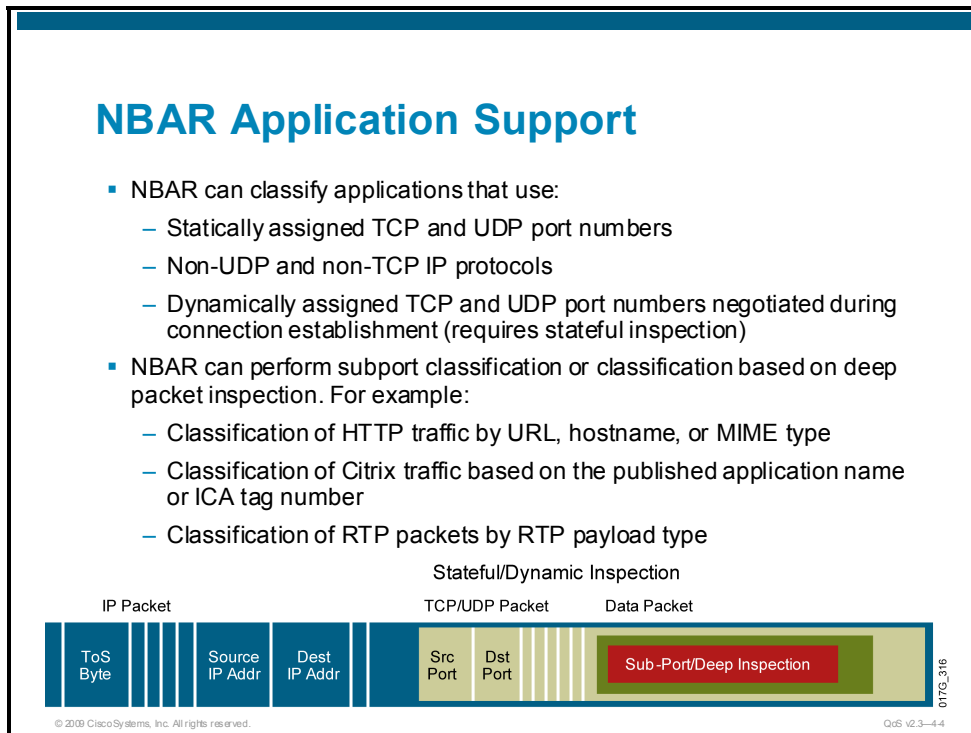
The phrase "Layer 2 NBAR" refers to NBAR functionality used with Layer 2 interfaces, such as switch ports, trunks, or EtherChannels. Layer 2 NBAR functionality can also be used with service modules such as a Firewall Services Module (FWSM) and an Intrusion Detection Services Module (IDSM) with the following restriction. Layer 2 NBAR is not supported on Layer 2 interfaces that are configured as part of a service module (such as FWSM and IDSM) when those service modules are configured in inline mode (that is, when network traffic is in a direct path through the service module). However, Layer 2 NBAR is supported in non-inline mode with service modules.

Note This restriction does not apply to NBAR functionality that is used with Layer 3 interfaces.

Note For the latest information regarding NBAR use restrictions, please refer to the Cisco IOS documentation for your specific software release.

NBAR Application Support

This topic describes the types of applications supported by NBAR.



NBAR supports simpler configuration coupled with stateful recognition of flows. The simpler configuration means that a protocol analyzer capture does not need to be examined to calculate ports and details. Stateful recognition means smarter, deeper packet recognition.

NBAR can be used to recognize and classify packets belonging to the following types of protocols and applications:

- **Applications that use statically assigned TCP and UDP port numbers:** These applications establish sessions to well-known TCP or UDP destination port numbers. Access control lists (ACLs) can also be used for classifying static port protocols. However, NBAR is easier to configure, and NBAR can provide classification statistics that are not available when ACLs are used.
- **Applications that use dynamically assigned TCP and UDP port numbers:** These applications use multiple sessions that use dynamic TCP or UDP port numbers. Typically, there is a control session to a well-known port number and the other sessions are established to destination port numbers negotiated through the control sessions. NBAR inspects the port number exchange through the control session. This kind of classification requires stateful inspection; that is, the ability to inspect a protocol across multiple packets during packet classification.
- **Non-TCP and non-UDP IP protocols:** Some non-TCP and non-UDP IP protocols can be recognized by NBAR.

NBAR also has the capability to perform subport classification or classification based on deep-packet inspection. Deep-packet classification is classification performed at a finer level of granularity. For instance, if a packet is already classified as HTTP traffic, it may be further classified as HTTP traffic with a specific URL. Subport classification looks beyond the TCP or UDP port numbers of a packet; NBAR looks into the TCP or UDP payload itself and classifies packets based on the content within the payload, such as transaction identifier, message type, or other similar data.

NBAR Application Support (Cont.)

TCP and UDP Protocols Supported by NBAR					
BGP	Finger	Kazaa	Novadigm	SFTP	SunRPC
Citrix ICA	FTP	Kerberos	NTP	SMTP	Syslog
CU-SeeMe	HTTP	L2TP	PCAnywhere	SNMP	Telnet
DHCP	HTTPS	LDAP	POP3	SOCKS	TFTP
DNS	IMAP	NetShow	RSVP	SQL*NET	VDOLive
Exchange	IRC	NNTP	RTSP	SSH	YouTube

Non-TCP and Non-UDP Protocols Supported by NBAR						
EGP	EIGRP	GRE	ICMP	IPINIP	IPsec	OSPF

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.6

The tables list some of the NBAR-supported protocols available in Cisco IOS Software. The tables also provide information about the protocol type and the well-known port numbers (if applicable).

Many peer-to-peer file-sharing applications not listed in these tables can be classified using FastTrack or Gnutella. RTSP can be used to classify various types of applications that use streaming audio.

Note These tables include the NBAR-supported protocols available with the 12.2(18)ZY and 12.2(18)ZYA releases. Protocols included in the 12.2(18)ZY and 12.2(18)ZYA releases are supported on the Catalyst 6500 series switch that is equipped with a Supervisor 32/PISA.

This table lists non-TCP and non-UDP protocols supported by NBAR.

Non-TCP and Non-UDP NBAR Supported Protocols

Protocol	Network Protocol	Protocol ID	Description
EGP	IP	8	Exterior Gateway Protocol
GRE	IP	47	Generic Routing Encapsulation
ICMP	IP	1	Internet Control Message Protocol
IPIP	IP	4	IP in IP
IPsec	IP	50, 51	IP Encapsulating Security Payload (ESP=50) and Authentication Header (AH=51)
EIGRP	IP	88	Enhanced Interior Gateway Routing Protocol
OSPF	IP	89	Open Shortest Path First

This table shows IP protocols supported by NBAR.

TCP and UDP NBAR Supported Protocols

Protocol	Network Protocol	Protocol ID	Description
AOL-messenger	TCP	5190, 443	AOL Instant Messenger chat messages
BGP	TCP/UDP	179	Border Gateway Protocol
Citrix ICA	TCP/UDP	TCP: 1494, 2512, 2513, 2598 UDP: 1604	Citrix ICA traffic
CU-SeeMe	TCP/UDP	TCP: 7648, 7649 UDP: 24032	Desktop video conferencing
DHCP/BOOTP	UDP	67, 68	Dynamic Host Configuration Protocol/ Bootstrap Protocol
DNS	TCP/UDP	53	Domain Name System
Doom	TCP/UDP	666	Doom
Exchange	TCP	135	MS-RPC for Exchange
FastTrack	TCP/UDP	Dynamically assigned	FastTrack peer-to-peer protocol
Finger	TCP	79	Finger user information protocol
FTP	TCP	Dynamically assigned, 20, 21	File Transfer Protocol
HTTP	TCP	80	Hypertext Transfer Protocol
HTTPS	TCP	443	Secure HTTP
IMAP	TCP/UDP	143, 220	Internet Message Access Protocol
IRC	TCP/UDP	194	Internet Relay Chat
Kazaa	TCP/UDP	Dynamically assigned	Kazaa
Kerberos	TCP/UDP	88, 749	Kerberos network authentication service
L2TP	UDP	1701	Layer 2 Tunneling Protocol
LDAP	TCP/UDP	389	Lightweight Directory Access Protocol
MSN-messenger	TCP	1863	MSN Messenger chat messages
NetShow	TCP/UDP	Dynamically assigned	Microsoft NetShow
NNTP	TCP/UDP	119	Network News Transfer Protocol
Notes	TCP/UDP	1352	Lotus Notes
Novadigm	TCP/UDP	3460-3465	Novadigm Enterprise Desktop Manager (EDM)
NTP	TCP/UDP	123	Network Time Protocol

Protocol	Network Protocol	Protocol ID	Description
PCAnywhere	TCP/UDP	TCP: 5631, 65301 UDP: 22, 5632	Symantec PCAnywhere
POP3	TCP/UDP	110	Post Office Protocol
RealAudio	TCP/UDP	Dynamically assigned	RealAudio Streaming Protocol
RSVP	UDP	1698,1699	Resource Reservation Protocol
RTSP	TCP/UDP	Dynamically assigned	Real Time Streaming Protocol
SFTP	TCP	990	Secure FTP
SIP	TCP/UDP	5060	Session Initiation Protocol
Skinny (SCCP)	TCP	2000, 2001, 2002	Skinny Client Control Protocol
Skype	TCP/UDP	Dynamically assigned	Peer-to-Peer VoIP Client Software
SMTP	TCP	25	Simple Mail Transfer Protocol
SNMP	TCP/UDP	161, 162	Simple Network Management Protocol
SOCKS	TCP	1080	Firewall security protocol
SQL*NET	TCP/UDP	1521	SQL*NET for Oracle
SSH	TCP	22	Secured Shell Protocol
SunRPC	TCP/UDP	Dynamically assigned	Sun Remote Procedure Call
Syslog	UDP	514	System logging utility
Telnet	TCP	23	Telnet protocol
TFTP	UDP	Static (69) with inspection	Trivial File Transfer Protocol
VDOLive	TCP/UDP	Static (7000) with inspection	VDOLive Streaming Video
Yahoo-messenger	TCP	5050, 5101	Yahoo Messenger chat messages
YouTube	TCP	Both static (80) and dynamically assigned	Online video-sharing website

Note For a complete list of NBAR-supported protocols (and details regarding protocol support with specific platforms and software versions), refer to the Classification section of the *Cisco IOS Quality of Service Solutions Configuration Guide*, Release 12.4 at <http://www.cisco.com>.

NBAR and Port Numbers Other Than Well-Known Ports

- Some protocols can use TCP or UDP ports other than the well-known IANA-assigned port numbers.
- You can configure NBAR to search for a protocol or protocol name using a port number other than the well-known port.

```
router(config)#
```

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

- Configures NBAR to search for a protocol or protocol name using a port number other than the well-known port
- Accepts up to 16 port-number arguments in one command line

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.6

Some protocols can use TCP or UDP ports other than the well-known Internet Assigned Numbers Authority (IANA)-assigned port number. You can configure NBAR to search for a protocol or protocol name using a port number other than the well-known port by using the **ip nbar port-map** command in global configuration mode. Using this command modifies the default NBAR port map, which associates protocols with their IANA-assigned port numbers. For example, you can use the **ip nbar port-map** command to configure NBAR to look for Telnet on a port other than port 23. You can specify up to 16 ports with this command.

Note For best results, do not configure the Citrix or BitTorrent protocols.

The syntax for the **ip nbar port-map** command is as follows:

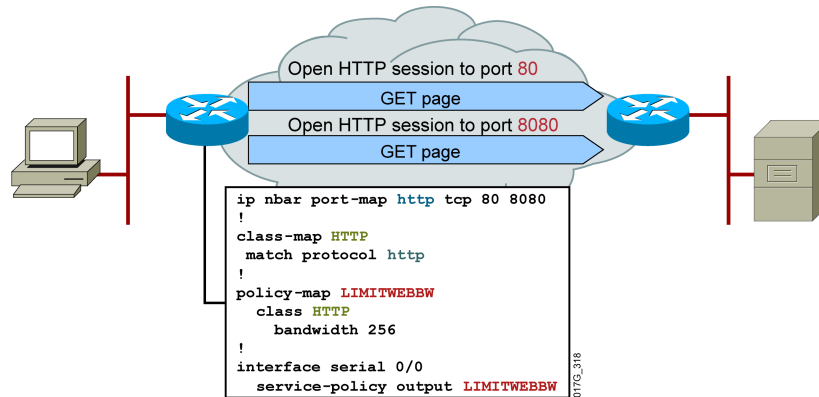
```
ip nbar port-map protocol-name [tcp | udp] port-number
```

Syntax Description

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

NBAR and Port Numbers Other Than Well-Known Ports (Cont.)

- HTTP typically uses well-known port number 80; however, other port numbers may also be used.
- The **ip nbar port-map** command informs the router that other ports are also used for HTTP.



© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.7

HTTP is often used on ports other than its well-known port, TCP port 80. In the figure, the **ip nbar port-map** command is used to enable HTTP recognition on both TCP port 80 and TCP port 8080. The class map called HTTP is used to match the HTTP protocol. The policy map LIMITWEBBW will use the class map HTTP and set the bandwidth for HTTP traffic to 256 kb/s. The policy map is then applied as a service policy for outbound traffic on the S0/0 interface.

The following example configures NBAR to look for the protocol Structured Query Language (SQL)*NET on port numbers 63000 and 63001, rather than on its well-known port numbers, TCP port 66 and UDP port 66:

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

You can use the **show ip nbar port-map** command to display the current protocol-to-port mappings in use by NBAR. Use the display to verify that these mappings are correct.

If the **ip nbar port-map** command has been used, the **show ip nbar port-map** command displays the ports assigned to the protocol. If the **no ip nbar port-map** command has been used, the **show ip nbar port-map** command displays the default ports. To limit the display to a specific protocol, use the **protocol-name** argument of the **show ip nbar port-map** command.

Packet Description Language Module

This topic describes the purpose of PDLMs in NBAR.

Packet Description Language Module

- An external PDLM can be loaded at run time to extend the NBAR list of recognized protocols.
- PDLMs allow NBAR to recognize new protocols without requiring a new Cisco IOS image or a router reload.
- PDLMs are produced exclusively by Cisco engineers.
- The DirectConnect PDLM is currently available for download.

© 2009 Cisco Systems, Inc. All rights reserved.

OID: 2.3-4.8

New features are usually added to new versions of the Cisco IOS Software. NBAR is the first mechanism that supports dynamic upgrades without having to change the Cisco IOS Software version or restart a router. This is accomplished by loading one or more PDLMs onto a router. Adding PDLMs extends the functionality of NBAR by enabling NBAR to recognize additional protocols on your network.

A PDLM is a separate file available on <http://www.cisco.com>. You can load an external PDLM at run time to extend the NBAR list of recognized protocols. PDLMs that are not embedded within Cisco IOS Software are referred to as nonnative PDLMs. A native PDLM is a PDLM that is embedded within the Cisco IOS Software. You receive it automatically along with the Cisco IOS Software.

There are separate version numbers associated with the NBAR software and the Cisco IOS Software. These version numbers are used together to maintain the PDLM version.

- **PDLM version:** The version of the PDLM, either native or nonnative.
- **Cisco IOS NBAR software version:** The version of NBAR that resides with the Cisco IOS Software. You can display the Cisco IOS NBAR software version by executing the **show ip nbar version** command.

Note Each nonnative PDLM also contains the Cisco IOS NBAR software version in which the PDLM was created.

Both the native and nonnative PDLMs contain internal module names. These internal module names are unique and independent. They are used to indicate the protocol that the PDLM module represents (such as BitTorrent or DirectConnect), and they are used to control the module version number (such as module version 3).

In order for a PDLM (either native or nonnative) to be downloaded or overridden, both of the following conditions must be met:

- The module version of the PDLM being downloaded must be higher than the module version currently installed. For example, if a PDLM called BitTorrent.pdlm (with the internal module name "bittorrent") is currently at version 3, the resident PDLM (either native or nonnative) with the same internal module name (bittorrent) is overridden as long as the module version is either 1 or 2.
- The Cisco IOS NBAR software version of the PDLM must be less than or equal to the Cisco IOS NBAR software version of the Cisco IOS image.

Note	New PDLMs are released only by Cisco and are available from local Cisco representatives. They can be loaded from flash memory. To view a list of currently available PDLMs, go to the following URL: http://www.cisco.com/pcgi-bin/tablebuild.pl/pdlm

NBAR and Custom Protocols

This topic explains how custom protocols can be used to extend the set of protocols that NBAR recognizes.

NBAR and Custom Protocols

- NBAR supports the use of custom protocols to identify custom applications.
- You can add to the set of protocols and application types that NBAR recognizes by creating custom protocols.
- You can define the names of your custom protocol applications; the custom protocol names you specify can then be used by Protocol Discovery, the Protocol Discovery MIB, the **match protocol** command, and the **ip nbar port-map** command as NBAR-supported protocols.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.8

NBAR supports the use of custom protocols to identify custom applications. You can add to the set of protocols and application types that NBAR recognizes by creating custom protocols.

NBAR recognizes and classifies network traffic on the basis of a set of protocols and application types. Custom protocols extend the capability of NBAR Protocol Discovery to classify and monitor additional static port applications and allows NBAR to classify nonsupported static port traffic; NBAR can map static TCP and UDP port numbers to custom protocols.

You can define the names of your custom protocol applications. Each name you specify can then be used by Protocol Discovery, the Protocol Discovery MIB, the **match protocol** command, and the **ip nbar port-map** command as an NBAR-supported protocol.

You can create more than 30 custom protocols on a router. NBAR can support up to 128 protocols total.

Protocol Discovery

This topic describes NBAR Protocol Discovery and the Cisco NBAR Protocol Discovery MIB.

Protocol Discovery

- Protocol Discovery analyzes application traffic patterns in real time and discovers which traffic is running on the network.
- Provides bidirectional, per-interface, per-protocol statistics:
 - 5-minute bit rate (b/s)
 - Packet counts
 - Byte counts
- Important monitoring tool supported by Cisco QoS management tools.
 - Generates real-time application statistics
 - Provides traffic distribution information at key network locations
- Historical QoS statistical information available through the Protocol Discovery MIB.

© 2009 Cisco Systems, Inc. All rights reserved.QoS v2.3-4-10

NBAR includes a Protocol Discovery feature that provides an easy way to discover application protocols that are transiting an interface so that appropriate QoS features can be applied. The Protocol Discovery feature discovers any protocol traffic that is supported by NBAR.

NBAR Protocol Discovery captures key statistics associated with each protocol in a network. These statistics can be used to define traffic classes and QoS policies for each traffic class.

You can apply protocol discovery to interfaces and use it to monitor both input and output traffic. In addition, protocol discovery shows the mix of applications currently running on the network, which helps in defining QoS classes and policies, such as determining how much bandwidth to provide to mission-critical applications and which protocols should be policed.

Protocol Discovery maintains the following per-protocol statistics for enabled interfaces:

- Total number of input packets and bytes
- Total number of output packets and bytes
- Input bit rates
- Output bit rates

Cisco IOS Software Release 12.2(18)ZYA includes a feature called Non-Intrusive Protocol Discovery. The Non-Intrusive Protocol Discovery feature enables the Cisco Catalyst 6500 Series Switch, which is equipped with a Supervisor 32/PISA, to perform protocol discovery in out-of-band (that is, offline) mode. In offline mode, a copy of the network traffic is used to discover the application protocols that are operating on an interface, leaving the network traffic undisturbed and available for other purposes. Non-Intrusive Protocol Discovery is closely

associated with a feature called Intelligent Traffic Redirect (ITR). ITR allows you to optimize system performance by identifying the specific traffic that needs to be redirected to the Supervisor 32/PISA for deep-packet inspection. Non-Intrusive Protocol Discovery is achieved by enabling ITR on an interface on which protocol discovery has been enabled.

For the Non-Intrusive Protocol Discovery feature to function properly, no other intrusive features (for example, Flexible Packet Matching [FPM]) can be in use on the interface in either the input or output direction. An intrusive feature is one that manipulates the packets, such as modifying a statistic or a packet counter. If such a feature is in use, the actual traffic (and not a copy of the traffic) is redirected.

Note	For more information about the commands used to enable ITR, see the <i>Cisco Catalyst Supervisor Engine 32 PISA IOS Command Reference</i> on http://www.cisco.com .

Protocol Discovery MIB

The NBAR Protocol Discovery MIB uses SNMP to provide the following protocol discovery functionality:

- Enable or disable protocol discovery per interface
- Display protocol discovery statistics
- Configure and view multiple top-n tables that list protocols by bandwidth usage
- Configure thresholds based on traffic of particular NBAR-supported protocols or applications that report breaches and send notifications when these thresholds are crossed

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3—4-11

The NBAR Protocol Discovery MIB expands the capabilities of NBAR Protocol Discovery by providing the following functionality through Simple Network Management Protocol (SNMP):

- Enable or disable Protocol Discovery per interface.
- Display Protocol Discovery statistics.
- Configure and view multiple top-n tables that list protocols by bandwidth usage.
- Configure thresholds based on traffic of particular NBAR-supported protocols or applications that report breaches and send notifications when these thresholds are crossed.

Configuring and Monitoring NBAR

This topic describes the Cisco IOS commands that are required to configure and monitor NBAR.

NBAR Configuration Tasks

1. Enable Cisco Express Forwarding (required)
2. Enable Protocol Discovery (as needed to determine which protocols are currently running on your network; required for obtaining statistics on those protocols)
3. Configure NBAR using the MQC (optional)
4. Add PDLs (optional)
5. Create custom protocols (optional)

To configuring NBAR, follow these steps:

- Step 1** Enable Cisco Express Forwarding (required unless you are using Cisco IOS Software Release 12.2(18)ZYA)
- Step 2** Enable Protocol Discovery (as needed to determine which protocols are currently running on your network; required for obtaining statistics on those protocols)
- Step 3** Configure NBAR using the MQC (optional)
- Step 4** Add PDLs (optional)
- Step 5** Creating custom protocols (optional)

Configuring Protocol Discovery

```
router(config-if)#
```

```
ip nbar protocol-discovery
```

- Configures NBAR to discover traffic for all protocols known to NBAR on a particular interface
- Requires that Cisco Express Forwarding be enabled before protocol discovery
- Can be applied with or without a service policy enabled

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-13

The NBAR feature has these two components:

- One component monitors applications traversing a network.
- The other component classifies traffic by protocol.

In order to monitor applications traversing a network, Protocol Discovery must be enabled. The ability to classify traffic by protocol using NBAR and then to apply QoS to the classified traffic is configured using the MQC.

Use the **ip nbar protocol-discovery** command in interface configuration mode (or VLAN configuration mode for Catalyst switches) to configure NBAR to keep traffic statistics for all protocols known to NBAR. The syntax shown in the figure for the **ip nbar protocol-discovery** command is the same as the syntax for the **ip nbar protocol-discovery** command in Cisco IOS Software Release 12.4.

Note	With Cisco IOS Software Release 12.2(18)ZYA, intended for use on a Cisco 6500 Series Switch that is equipped with a Supervisor 32/PISA, the ip nbar protocol-discovery command is supported on both Layer 2 and Layer 3 EtherChannels.

Monitoring Protocol Discovery

router#

```
show ip nbar protocol-discovery [interface type number]
[stats {byte-count | bit-rate | packet-count | max-bit-
rate}] [protocol protocol-name] [top-n number]
```

- Displays the statistics gathered by the NBAR Protocol Discovery feature

© 2009 Cisco Systems, Inc. All rights reserved.

QoS V2.3-4-14

Use the **show ip nbar protocol-discovery** command to display statistics gathered by the NBAR protocol discovery feature. This command, by default, displays statistics for all interfaces on which protocol discovery is currently enabled.

You can use protocol discovery to monitor both input and output traffic and apply it with or without a service policy enabled. NBAR protocol discovery gathers statistics for packets switched to output interfaces. These statistics are not necessarily for packets that exited the router on the output interfaces, because packets might have been dropped after switching for various reasons (policing at the output interface, access lists, or queue drops).

The syntax for the **show ip nbar protocol-discovery** command in Cisco IOS Software Release 12.4 is as follows:

```
show ip nbar protocol-discovery [interface type number] [stats
{byte-count | bit-rate | packet-count | max-bit-rate}]
[protocol protocol-name] [top-n number]
```

Syntax Description

Parameter	Description
interface	(Optional) Specifies that protocol discovery statistics for the interface are to be displayed
<i>type</i>	Type of interface or subinterface whose policy configuration is to be displayed
<i>number</i>	Port, connector, VLAN, or interface card number
stats	(Optional) Specifies that the byte count, byte rate, or packet count is to be displayed
byte-count	(Optional) Specifies that the byte count is to be displayed
bit-rate	(Optional) Specifies that the bit rate is to be displayed
packet-count	(Optional) Specifies that the packet count is to be displayed
max-bit-rate	(Optional) Specifies that the maximum bit rate is to be displayed
protocol	(Optional) Specifies that statistics for a specific protocol are to be displayed
<i>protocol-name</i>	(Optional) User-specified protocol name for which the statistics are to be displayed
top-n	(Optional) Specifies that a top-n is to be displayed. A top-n is the number of most active NBAR-supported protocols, where n is the number of protocols to be displayed. For instance, if top-n 3 is entered, the three most active NBAR-supported protocols will be displayed.
<i>number</i>	(Optional) Specifies the number of most active NBAR-supported protocols to be displayed

Monitoring Protocol Discovery (Cont.)

```
router# show ip nbar protocol-discovery top-n 5

Ethernet2/0

Protocol                                Input                                Output
-----                                -
Packet Count                            Packet Count
Byte Count                              Byte Count
5min Bit Rate (bps)                     5min Bit Rate (bps)
5min Max Bit Rate (bps)                 5min Max Bit Rate (bps)
-----
rtp                                     3272685                             3272685
                                     242050604                           242050604
                                     768000                               768000
                                     2002000                              2002000
gnutella                              513574                              513574
                                     118779716                            118779716
                                     383000                               383000
                                     987000                               987000
ftp                                    482183                              482183
. . .
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS V2.3-4-15

The default output of the **show ip nbar protocol-discovery** command includes input bit rate (in bits per second), input byte count, input packet count, and protocol name. The example in the figure displays partial output from the **show ip nbar protocol-discovery** command for the five most active protocols on an Ethernet interface.

Configuring NBAR Using the MQC

```
router(config-cmap)#
```

```
match protocol protocol
```

- Configures the match criteria for a class map on the basis of the specified protocol.

```
class-map CMAP1
  match protocol citrix
!
policy-map POLICY1
  class CMAP1
    bandwidth percent 50
!
interface FastEthernet2/4
  service-policy input POLICY1
```

- CBWFQ is applied to all Citrix traffic entering the FastEthernet2/4 interface.

© 2009 Cisco Systems, Inc. All rights reserved.QoS v2.3-4-10

After you enable Protocol Discovery, you can configure NBAR using the functionality of the MQC. The MQC uses traffic classes and traffic policies (policy maps) to apply QoS features to classes of traffic and applications recognized by NBAR. Configuring NBAR using the MQC involves defining a traffic class, configuring a traffic policy (policy map), and then attaching that traffic policy to the appropriate interface.

A traffic class contains three major elements: a name, one or more **match** commands, and, if more than one **match** command exists in the traffic class, an instruction on how to evaluate these **match** commands (**match-all** or **match-any**).

The **match** commands are used to specify various criteria for classifying packets. For NBAR, the criterion is whether the packet matches a specific protocol or application known to NBAR. Using the MQC, network traffic with one network protocol (Citrix, for example) can be placed into one traffic class, while traffic that matches a different network protocol (Gnutella, for example) can be placed into another traffic class. You specify the criteria used to classify traffic by using the **match protocol** command. Using the **match protocol** command with the *protocol-name* value configures NBAR to match all traffic for the specified protocol.

Note For Cisco IOS Software Release 12.2(18)ZY, a maximum of eight **match protocol** commands can be configured in a single traffic class.

In the figure, a class called CMAP1 is configured to match all Citrix traffic. CMAP1 is applied to policy map POLICY1, within which the **bandwidth** command is used to associate CBWFQ with the CMAP1 class of traffic. POLICY1 is attached to Ethernet interface 2/4 in the input direction. CBWFQ is an example of a QoS feature that can be applied in a policy map. Use the appropriate command for the QoS feature that you want to use.

The syntax for the **match protocol** command is as follows:

match protocol *protocol-name*

Syntax Description

Parameter	Description
<i>protocol-name</i>	Name of the protocol used as a matching criterion. Supported protocols include the following (Some protocols omitted. Refer to Cisco IOS documentation for complete details): aarp : AppleTalk Address Resolution Protocol (AARP) arp : IP Address Resolution Protocol (ARP) bridge : bridging cdp : Cisco Discovery Protocol compressedtcp : compressed TCP dls : data-link switching (DLSw) ip : IP ipx : Novell IPX

The syntax for the **match protocol** command in Cisco IOS Software Release 12.4 is as follows:

```
match protocol protocol-name [variable-field-name value]
```

Syntax Description

Parameter	Description
<i>protocol-name</i>	Particular protocol type that is known to NBAR.
<i>variable-field-name</i>	(Optional and usable only with custom protocols) Predefined variable that was created when you created a custom protocol. The <i>variable-field-name</i> will match the <i>field-name</i> variable entered when you created the custom protocol using the ip nbar custom command.
<i>value</i>	(Optional and usable only with custom protocols) Specific value in the custom payload to match. A value can be entered along with a variable field name. The value can be expressed in decimal or hexadecimal format.

Classifying HTTP Traffic

```
router(config-cmap) #
```

```
match protocol http [url url-string | host hostname-string | mime  
MIME-type | c-header-field c-header-field-string | s-header-field  
s-header-field-string]
```

- Configures NBAR to match HTTP traffic by URL, host, MIME type, or fields in HTTP packet headers

```
class-map CLASS2  
match protocol http url whatsnew/latest*
```

- Classifies HTTP packets based on any URL containing the string *whatsnew/latest* followed by zero or more characters

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-17

NBAR can classify application traffic by looking beyond the TCP and UDP port numbers of a packet. This capability is called subport classification. NBAR looks into the TCP or UDP payload itself and classifies packets based on content within the payload, such as transaction identifier or message type. Classification of HTTP traffic by URL, host, or MIME type is an example of subport classification.

NBAR classifies HTTP traffic by text within the URL or host fields of an HTTP request packet by using regular expression matching. When matching by URL, NBAR recognizes the HTTP packets containing the URL and then matches all packets that are part of the HTTP request. HTTP client request matching in NBAR supports most HTTP request methods such as GET, PUT, HEAD, POST, DELETE, OPTIONS, CONNECT, and TRACE. With classification by host name, NBAR classifies all packets from the host you specify.

For MIME-type matching, the MIME type can contain any text string you specify. A list of the IANA-supported MIME types can be found at the following URL:

<http://www.iana.org/assignments/media-types/>

NBAR supports URL and host classification in the presence of persistent HTTP. NBAR does not classify packets that are part of a pipelined request. With pipelined requests, multiple requests are pipelined to the server before previous requests are serviced. Pipelined requests are less common types of persistent HTTP requests.

In Cisco IOS Software Release 12.3(4)T, the NBAR Extended Inspection for HTTP Traffic feature was introduced. This feature allows NBAR to scan TCP ports that are not well known and to identify HTTP traffic traversing these ports. HTTP traffic classification is no longer limited to the well-known and defined TCP ports. This feature is enabled automatically when a service policy containing the **match protocol http** command is attached to an interface.

The syntax for the **match protocol http** command in Cisco IOS Software Release 12.4 is as follows:

```
match protocol http [url url-string | host hostname-string |  
mime MIME-type | c-header-field c-header-field-string | s-  
header-field s-header-field-string]
```

Parameter	Description
url	(Optional) Specifies matching by a URL
<i>url-string</i>	(Optional) User-specified URL of HTTP traffic to be matched
host	(Optional) Specifies matching by a hostname
<i>hostname-string</i>	(Optional) User-specified hostname to be matched
mime	(Optional) Specifies matching by a MIME text string
<i>MIME-type</i>	(Optional) User-specified MIME text string to be matched
c-header-field	(Optional) Specifies matching by a string in the header field in HTTP request messages
<i>c-header-field-string</i>	(Optional) User-specified text string within the HTTP request message to be matched
s-header-field	(Optional) Specifies matching by a string in the header field in HTTP response messages
<i>s-header-field-string</i>	(Optional) User-specified text within the HTTP response message to be matched

When specifying a URL for classification, include only the portion of the URL that follows the `www.hostname.domain` in the match statement. For example, for the URL `www.cisco.com/latest/whatsnew.html`, include only `/latest/whatsnew.html` with the match statement as shown in the following example:

```
class-map EXAMPLECLASS1  
  match protocol http url /latest/whatsnew.html
```

The following example classifies, within the class map called “EXAMPLECLASS2,” packets based on any host name containing the string “cisco” followed by zero or more characters:

```
class-map EXAMPLECLASS2  
  match protocol http host cisco*
```

The following example classifies, within the class map called “EXAMPLECLASS3,” packets based on the Joint Photographic Experts Group (JPEG) MIME type:

```
class-map EXAMPLECLASS3  
  match protocol http mime "*jpeg"
```

Classifying HTTP Traffic (Cont.)

```
class-map match-all CLASS1
match protocol http c-header-field "somebody@cisco.com"
```

- Any request message that contains "somebody@cisco.com" in the User-Agent, Referrer, or From fields will be classified by NBAR.

```
class-map match-all C-HTTP
match protocol http c-header-field "CERN-LineMode/3.0"
match protocol http s-header-field "CERN/3.0"
match protocol http url "www.cisco.com/routers"
```

- HTTP header fields are combined with a URL to classify traffic.
- Traffic with a User-Agent field of "CERN-LineMode/3.0" and a Server field of "CERN/3.0," along with URL "www.cisco.com/routers," will be classified using NBAR.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-18

In Cisco IOS Software Release 12.3(11)T, NBAR introduced expanded ability for users to classify HTTP traffic using information in the HTTP header fields. HTTP header fields are used to provide information about HTTP request and response messages. HTTP has numerous header fields.

NBAR is able to classify these HTTP header fields for request messages (client to server):

- User-Agent
- Referer
- From

NBAR is able to classify the following HTTP header fields for response messages (server to client):

- Server
- Location
- Content-Encoding
- Content-Base

Note Use of the Content-Base field has not been implemented by the HTTP community. (See RFC 2616 for details.) Therefore, the Content-Base field is not identified by NBAR on the Cisco Catalyst 6500 Series Switch that is equipped with a Supervisor 32/PISA.

You can use the **match protocol http** command with the **c-header-field** keyword to specify that NBAR identify request messages. The "c" in the **c-header-field** portion of the command is for client. You can use the **match protocol http** command with the **s-header-field** keyword to specify response messages. The "s" in the **s-header-field** portion of the command is for server.

In the first example in the figure, any request message that contains "somebody@cisco.com" in the User-Agent, Referer, or From fields will be classified by NBAR. Typically, a term with a format similar to "somebody@cisco.com" would be found in the From header field of the HTTP request message.

Classifying Citrix Traffic

```
router(config-cmap) #
```

```
match protocol citrix [app application-name-string] [ica-tag ica-tag-value]
```

- Configures NBAR to match Citrix traffic

```
class-map CLASS3  
match protocol citrix app packet1
```

- Configures NBAR to match Citrix traffic with the application name of packet1

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-19

NBAR can classify Citrix Independent Computing Architecture (ICA) traffic and perform subport classification of Citrix traffic based on the published application name or ICA tag number. To configure NBAR to match Citrix traffic, use the **match protocol citrix** command in class-map configuration mode.

The syntax for the **match protocol citrix** command in Cisco IOS Software Release 12.4 is as follows:

```
match protocol citrix [app application-name-string] [ica-tag ica-tag-value]
```

Syntax Description

Parameter	Description
app	(Optional) Specifies matching of an application name string
<i>application-name-string</i>	(Optional) Specifies the string to be used as the subprotocol parameter
ica-tag	(Optional) Specifies tagging of Independent Computing Architecture (ICA) packets
<i>ica-tag-value</i>	(Optional) Specifies the priority tag of ICA packets. Priority tag values can be in the range of 0 to 3

Note For more information on NBAR and Classification of Citrix ICA Traffic, refer to the Part 1: Classification section of the *Cisco IOS Quality of Service Solutions Configuration Guide*, Release 12.4 on <http://www.cisco.com>.

Classifying Peer-to-Peer File-Sharing Applications

```
router(config-cmap) #
```

```
match protocol gnutella file-transfer "regular-expression"
```

- Configures NBAR to match Gnutella peer-to-peer traffic

```
router(config-cmap) #
```

```
match protocol fasttrack file-transfer "regular-expression"
```

- Configures NBAR to match FastTrack peer-to-peer traffic

© 2009 Cisco Systems, Inc. All rights reserved.

QoS V2.3-4-20

The following are the most common peer-to-peer file-sharing applications supported by NBAR:

- BitTorrent
- DirectConnect
- eDonkey
- eMule
- FastTrack
- Grokster
- JTella
- Kazaa (as well as Kazaa Lite and Kazaa Lite Resurrection)
- Morpheus
- Win MX

Gnutella is another file-sharing protocol that became classifiable using NBAR in Cisco IOS Software Release 12.1(12c)E. Applications that use the Gnutella protocol include Bearshare, Gnewtellium, Gnucleus, Gtk-Gnutella, Limewire, Mutella, Phex, Qtella, Swapper, and Xolo.

The **match protocol gnutella file-transfer** *regular-expression* and **match protocol fasttrack file-transfer** *regular-expression* commands are used to enable Gnutella and FastTrack classification in a traffic class. The **file-transfer** keyword indicates that a regular expression variable will be used to identify specific Gnutella or FastTrack traffic. Wildcard characters in a regular expression can also be used to identify specified Gnutella and FastTrack traffic. These regular expression matches can be used to match on the basis of filename extension or a particular string in a filename. In the example, Gnutella traffic that contains the characters "cisco" is classified:

```
class-map match-all NBAR
```

```
match protocol gnutella file-transfer "*cisco*"
```

Classifying Peer-to-Peer File-Sharing Applications (Cont.)

```
class-map match-all NBAR
match protocol gnutella file-transfer "*.mpeg"
```

- All Gnutella files that have the .mpeg extension will be classified into class map NBAR

```
class-map match-all NBAR
match protocol fasttrack file-transfer "*"
```

- All FastTrack traffic is classified into class map NBAR

```
class-map match-all NBAR
match protocol edonkey
```

- All eDonkey and eMule traffic is classified into class map NBAR.
(NBAR also classifies eMule traffic as eDonkey traffic.)

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-21

The regular-expression variable can be expressed as "*" to indicate that all FastTrack or Gnutella traffic is classified by a traffic class. In the following example, all FastTrack traffic is classified into class map NBAR:

```
class-map match-all NBAR
match protocol fasttrack file-transfer "*"
```

Similarly, all Gnutella traffic is classified into class map NBAR in the following example:

```
class-map match-all NBAR
match protocol gnutella file-transfer "*"
```

To classify NBAR supported peer-to-peer file-sharing applications other than Gnutella and FastTrack, use the **match protocol** command as you would with other NBAR supported protocols.

Adding PDLMs

```
router(config)#
```

```
ip nbar pdlm pdlm-name
```

- Extends the list of protocols recognized by NBAR by installing a PDLM
- Uses URL format to specify the name and location of the PDLM file

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

- Downloads the DirectConnect PDLM to the router from flash memory

© 2009 Cisco Systems, Inc. All rights reserved.QoS V2.3-4-22

A PDLM is used to add support for a protocol that is currently not available as part of the Cisco IOS Software. A PDLM extends the functionality of NBAR by enabling NBAR to recognize additional protocols on the network.

To install a PDLM, complete these steps:

- Step 1** Review the list of NBAR protocols currently supported by the Cisco IOS Software release that you are using to determine if support for the protocol that you would like to add is already in your Cisco IOS Software release. Protocols introduced when you download a PDLM are commonly added to subsequent Cisco IOS Software releases. To check the list of NBAR protocols supported in your Cisco IOS Software release, enter the **match protocol ?** command and view the options that are displayed. The options correspond to the NBAR supported protocols.
- Step 2** If support for the protocol that you would like to add does not already exist in your Cisco IOS Software release, use the **ip nbar pdlm** command followed by the *pdlm-name* argument to add NBAR support for the protocol to your Cisco IOS Software. For the *pdlm-name* argument, enter the URL at which the PDLM can be found on the flash card.

Verifying Downloaded PDLMs

router#

```
show ip nbar pdlm
```

- Displays the PDLM in use by NBAR

```
router#show ip nbar pdlm
The following PDLMs have been loaded:
flash://citrix.pdlm
```

router#

```
show ip nbar version [pdlm-name]
```

- Displays information about the version of the NBAR software in your Cisco IOS Software release or the version of an NBAR PDLM on your router

```
router#show ip nbar version http
http                      Mv: 7, Nv: 3; slot1:http_vers.pdlm
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-23

After you download the PDLM, you may want to verify that the PDLM is on your router. You may also want to determine if there are earlier versions of the PDLM already on your router. You can use the output of the **show ip nbar pdlm** command and the **show ip nbar version** command to verify information about the downloaded PDLMs on your router.

In this example, the Citrix PDLM has been loaded from flash memory:

```
router# show ip nbar pdlm
The following PDLMs have been loaded:
flash://citrix.pdlm
```

The following is sample output from the **show ip nbar version** command:

```
router# show ip nbar version
NBAR software version: 3
 1  base                      Mv: 2
 2  ftp                       Mv: 2
 3  http                      Mv: 7, Nv: 3; slot1:http_vers.pdlm
 4  static-port               Mv: 6
 5  tftp                      Mv: 1
 6  exchange                  Mv: 1
 7  vdolive                    Mv: 1
 8  sqlnet                     Mv: 1
 9  rcmd                       Mv: 1
10  netshow                    Mv: 1
11  sunrpc                     Mv: 2
12  streamwork                 Mv: 1
```


13	citrix	Mv: 5
14	napster	Mv: 2
15	fasttrack	Mv: 2
16	gnutella	Mv: 1
17	kazaa	Mv: 6, Nv: 3; slot1:kazaa2_vers.pdlm
18	custom-protocols	Mv: 1
19	rtsp	Mv: 1
20	rtp	Mv: 2
21	mgcp	Mv: 1
22	skinny	Mv: 1
23	h323	Mv: 1
24	sip	Mv: 1
25	rtcp	Mv: 1

The table describes the fields shown in the display.

Field Descriptions for show ip nbar version Command Output

Field	Description
NBAR software version	NBAR software version that is running in the current Cisco IOS Software. In this example, version 3 is shown.
Mv	Resident Module Version. The Resident Module Version is the version of the NBAR-supported PDLM protocol, and therefore, varies by protocol. The Resident Module Version of TFTP, for example, is 1.
Nv	Minimum version of the NBAR software that is required to load a nonnative PDLM. This number is available only for nonnative PDLMs that were loaded onto the router, such as the Kazaa PDLM (protocol 17); in that case, the Nv version is 3.

Configuring Custom Protocols

```
router(config)#
```

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

- Defines a custom protocol

```
router(config)# ip nbar custom LunarLight 8 ascii Moonbeam  
tcp range 2000 2999
```

- Defines a custom protocol named LunarLight that identifies TCP packets that have source or destination ports between 2000 and 2999 (inclusive) and contain the string "Moonbeam" in the eighth byte of the payload

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-25

To define a custom protocol, use the keywords and arguments of the **ip nbar custom** command. After you define the custom protocol, you must create a traffic class and configure a traffic policy (policy map) to use the custom protocol when NBAR classifies traffic.

The syntax for the **ip nbar custom** command in Cisco IOS Software Release 12.4 is as follows:

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

Syntax Description

Parameter	Description
<i>name</i>	The name given to the custom protocol. This name is reflected wherever the name is used, including NBAR Protocol Discovery, the match protocol command, the ip nbar port-map command, and the NBAR Protocol Discovery MIB. The name must be no longer than 24 characters and can contain only lowercase letters (a-z), digits (0-9), and the underscore (_) character.
<i>offset</i>	(Optional) A digit representing the byte location for payload inspection. The offset function is based on the beginning of the payload directly after the TCP or UDP header.
<i>format value</i>	(Optional) Defines the format of the value and the length of the value that is being inspected in the packet payload. Current format options are ascii, hex, and decimal. The length of the value is dependent on the chosen format. The length restrictions for each format are listed below: ■ ascii: Up to 16 characters can be searched. Regular expressions are not supported. ■ hex: Up to 4 bytes. ■ decimal: Up to 4 bytes.
variable <i>field-name</i> <i>field-length</i>	(Optional) When you enter the variable keyword, a specific portion of the custom protocol can be treated as an NBAR-supported protocol. For example, a specific portion of the custom protocol can be tracked using class-map statistics and can be matched using the class-map command. If you enter the variable keyword, you must define the following fields: ■ field-name: Provides a name for the field to search in the payload. After you configure a custom protocol using a variable, you can use this field name with up to 24 different values per router configuration. ■ field-length: Enters the field length in bytes. The field length can be up to 4 bytes, so you can enter 1, 2, 3, or 4 as the field-length value.
source / destination	(Optional) Specifies the direction in which packets are inspected. If you do not specify source or destination, all packets traveling in either direction are monitored by NBAR.
tcp udp	(Optional) Specifies the TCP or the UDP implemented by the application.
range <i>start end</i>	(Optional) Specifies a range of ports that the custom application monitors. The <i>start</i> is the first port in the range, and the <i>end</i> is the last port in the range. One range of up to 1000 ports can be specified for each custom protocol.
<i>port-number</i>	(Optional) The port that the custom application monitors. Up to 16 individual ports can be specified as a single custom protocol.

In the following example, the custom protocol AppSales1 identifies TCP packets that have a source port of 4567 and that contain the term "SALES" in the fifth byte of the payload:

```
router(config)# ip nbar custom AppSales1 5 ascii SALES source
tcp 4567
```

In the following example, the custom protocol VirusHome identifies UDP packets that have a destination port of 3000 and that contain "0x56" in the seventh byte of the payload:

```
router(config)# ip nbar custom VirusHome 7 hex 0x56
destination udp 3000
```

In the following example, the custom protocol MediaNew identifies TCP packets that have a destination or source port of 4500 and that have a value of 90 at the sixth byte of the payload:

```
router(config)# ip nbar custom MediaNew 6 decimal 90 tcp 4500
```

In the following example, the custom protocol MSN1 looks for TCP packets that have a destination or source port of 6700:

```
router(config)# ip nbar custom MSN1 tcp 6700
```

In the following example, the custom protocol MailX looks for UDP packets that have a destination port of 8202:

```
router(config)# ip nbar custom MailX destination udp 8202
```

In the following example, the custom protocol MailY looks for UDP packets that have destination ports between 3000 and 4000 inclusive:

```
router(config)# ip nbar custom MailY destination udp range
3000 4000
```

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

```
router(config)# ip nbar custom FTDD 23 variable scid 1 tcp
range 5001 5005
```

```
router(config)# class-map ActiveCraft
router(config-cmap)# match protocol FTDD scid 0x15
router(config-cmap)# match protocol FTDD scid 0x21
router(config-cmap)# match protocol FTDD scid 0x27
```

```
router(config)# class-map PassiveCraft
router(config-cmap)# match protocol ftdd scid 0x11
router(config-cmap)# match protocol ftdd scid 0x22
router(config-cmap)# match protocol ftdd scid 0x25
```

Configuring RTP Payload Type Classification

```
router(config-cmap)#
```

```
match protocol rtp [audio | video | payload-type payload-string]
```

- Configures NBAR to match RTP traffic
- Can match audio traffic by payload-type values 0 to 23, which are reserved for audio traffic
- Can match video traffic by payload-type values 24 to 33, which are reserved for video traffic
- Supports NBAR RTP payload classification:
 - Allows you to identify real-time audio and video traffic
 - Can differentiate on the basis of audio and video codecs to provide more granular QoS

```
class-map RTPTRAFFIC
```

```
match protocol rtp payload-type "0, 1, 64"
```

- Configures NBAR to match RTP traffic with payload types 0, 1, and 64

© 2009 Cisco Systems, Inc. All rights reserved.

QoS V2.3-4.36

Real-Time Transport Protocol (RTP) is a packet format for multimedia data streams. It can be used for media-on-demand as well as for interactive services such as Internet telephony.

RTP consists of a data component and a control component. The control component is called Real-Time Transport Control Protocol (RTCP). RTCP is a separate protocol that is supported by NBAR. The data component of RTP is a thin protocol that provides support for applications with real-time properties such as continuous media (audio and video), which includes timing reconstruction, loss detection, and security and content identification.

The RTP payload is the data transported by RTP in a packet, such as audio samples or compressed video data. NBAR RTP Payload Type Classification not only allows you to statefully identify real-time audio and video traffic but can also differentiate on the basis of audio and video codecs to provide more granular QoS. The RTP Payload Type Classification feature, therefore, looks deep into the RTP header to classify RTP packets.

NBAR matches RTP traffic on the basis of the payload type field in the RTP packet. A working knowledge of RTP and RTP payload types is helpful if you want to configure NBAR to match RTP traffic. For more information about RTP and RTP payload types, refer to RFC 1889, *RTP: A Transport Protocol for Real-Time Applications*.

Note It is important to note that the NBAR RTP Payload Classification feature does not identify RTCP packets.

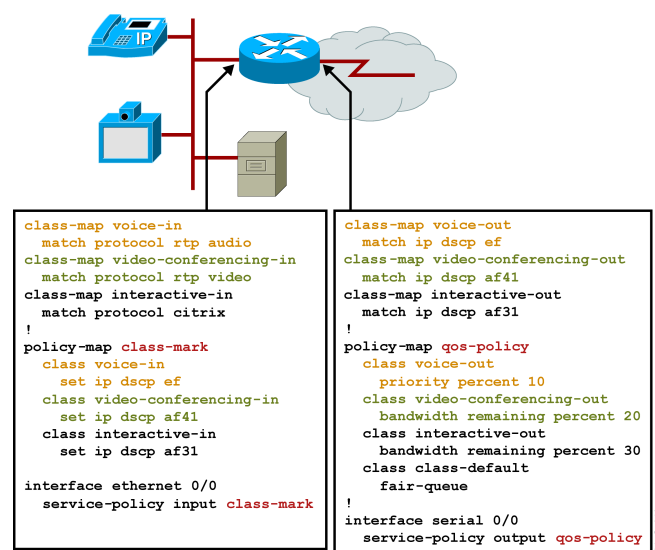
Use the **match protocol rtp** command to configure NBAR to match RTP traffic, including matching on RTP payload type. The syntax for the **match protocol rtp** command is as follows:

```
match protocol rtp [audio | video | payload-type payload-string]
```

Syntax Description

Parameter	Description
audio	(Optional) Specifies matching by audio payload-type values in the range of 0 to 23. These payload-type values are reserved for audio traffic.
video	(Optional) Specifies matching by video payload-type values in the range of 24 to 33. These payload-type values are reserved for video traffic.
payload-type	(Optional) Specifies matching by a specific payload-type value, providing more granularity than is available with the audio or video keywords.
<i>payload-string</i>	(Optional) User-specified string that contains the specific payload-type values. A <i>payload-string</i> argument can contain commas to separate payload-type values and hyphens to indicate a range of payload-type values. A <i>payload-string</i> argument can be specified in hexadecimal notation (by prepending 0x to the value) and binary notation (by prepending b to the value) in addition to standard number values.

Example: Configuring NBAR



The example in the figure illustrates a simple classification of RTP sessions, both on the input interface and on the output interface of the router. On the input interface, three class maps have been created: voice-in, videoconferencing-in, and interactive-in. The voice-in class map will match the RTP audio protocol; the videoconferencing-in class map will match the RTP video protocol; and the interactive-in class map will match the Citrix protocol.

The policy map class-mark will then do the following:

- If the packet matches the voice-in class map, the packet differentiated services code point (DSCP) field will be set to Expedited Forwarding (EF). If the packet matches the videoconferencing-in class map, the packet DSCP field will be set to AF41. If the packet matches the interactive-in class map, the DSCP field will be set to AF31.
- The policy map class mark is applied to the input interface, E0/0.

On the output interface, three class maps have been created: voice-out, videoconferencing-out, and interactive-out. The voice-out class map will match the DSCP field for EF. The videoconferencing-out class map will match the DSCP field for AF41. The interactive-out class map will match the DSCP field for AF31.

In the figure, policy-map qos-policy will then do the following:

- If the packet matches the class map voice-out, the LLQ priority bandwidth will be set to 10 percent of the interface bandwidth. If the packet matches the class map videoconferencing-out, the class-based weighted-fair queuing (CBWFQ) minimum-guaranteed bandwidth will be set to 20 percent of the interface bandwidth and if the packet matches the class map interactive-out, the CBWFQ will be set to 30 percent. All other packet flows will be classified as class-default and fair-queuing will be performed on them.
- The policy map class-mark is applied to the output interface, S0/0.

Note Your Cisco IOS Software release may not support all of the features documented in this lesson. For the latest feature information and caveats, see the release notes for your platform and software release. Use Cisco Feature Navigator to find information about platform support and Cisco IOS and Catalyst operating system image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>.

Note All commands shown in this lesson may not be available in your Cisco IOS Software release. For information about a specific command, see the command reference documentation.

Summary

This topic summarizes the key points discussed in this lesson.

Summary

- NBAR identifies applications and protocols (Layer 4 to Layer 7) and provides traffic statistics.
- NBAR recognizes applications that use statically assigned TCP and UDP port numbers, applications that use dynamically assigned TCP and UDP port numbers, and some non-TCP and non-UDP IP protocols.
- PDLs contain the rules that are used by NBAR to recognize an application and can be used to bring new or changed functionality to NBAR.
- You can configure NBAR to recognize custom protocols.
- NBAR Protocol Discovery analyzes application traffic patterns in real time and discovers which traffic is running on the network. NBAR Protocol Discovery MIB uses SNMP to provide new protocol discovery functionality.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-38

Summary (Cont.)

- You can use the following commands to configure NBAR:
 - **ip nbar protocol-discovery**: To configure NBAR to keep traffic statistics for protocols
 - **match protocol** (and variations): To specify a protocol as criteria used to classify traffic in a class map
 - **ip nbar pdlm**: To add support for a protocol that is currently not available as part of the Cisco IOS Software
 - **ip nbar custom**: To create a custom protocol
 - **ip nbar port-map**: To configure NBAR to search for a protocol using a port number other than the well-known port

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-39

Configuring QoS Preclassify

Overview

The quality of service (QoS) for virtual private networks (VPNs) feature (QoS preclassify) provides a solution for ensuring that Cisco IOS QoS services operate in conjunction with tunneling and encryption on an interface. This lesson describes QoS preclassify, using QoS policies on VPN interfaces, and configuring and monitoring QoS preclassify.

Objectives

Upon completing this lesson, you will be able to use the QoS preclassify feature to classify Generic Routing Encapsulation (GRE) and IPsec encapsulated packets. This ability includes being able to meet these objectives:

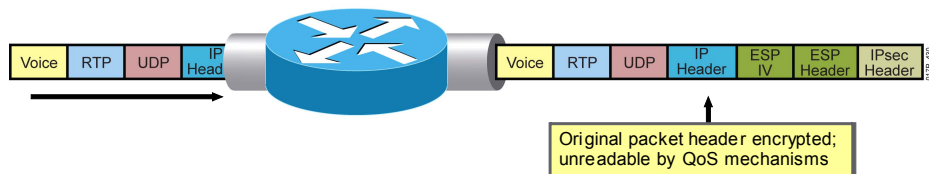
- Describe issues regarding the use of QoS with VPNs
- Describe ToS byte preservation
- Describe the QoS preclassification feature of Cisco IOS Software
- Identify the Cisco IOS commands required to support QoS preclassification
- Identify the Cisco IOS commands used to monitor QoS preclassification

QoS Issues with VPNs

This topic describes issues regarding the use of QoS with VPNs.

VPN QoS Issues

- Packet classification is commonly based on the contents of fields in the packet header.
- When packets are encrypted, QoS features are unable to examine the original packet headers and correctly classify packets.
- Packets traveling across the same tunnel have the same tunnel headers, so the packets are treated identically if the physical interface is congested.



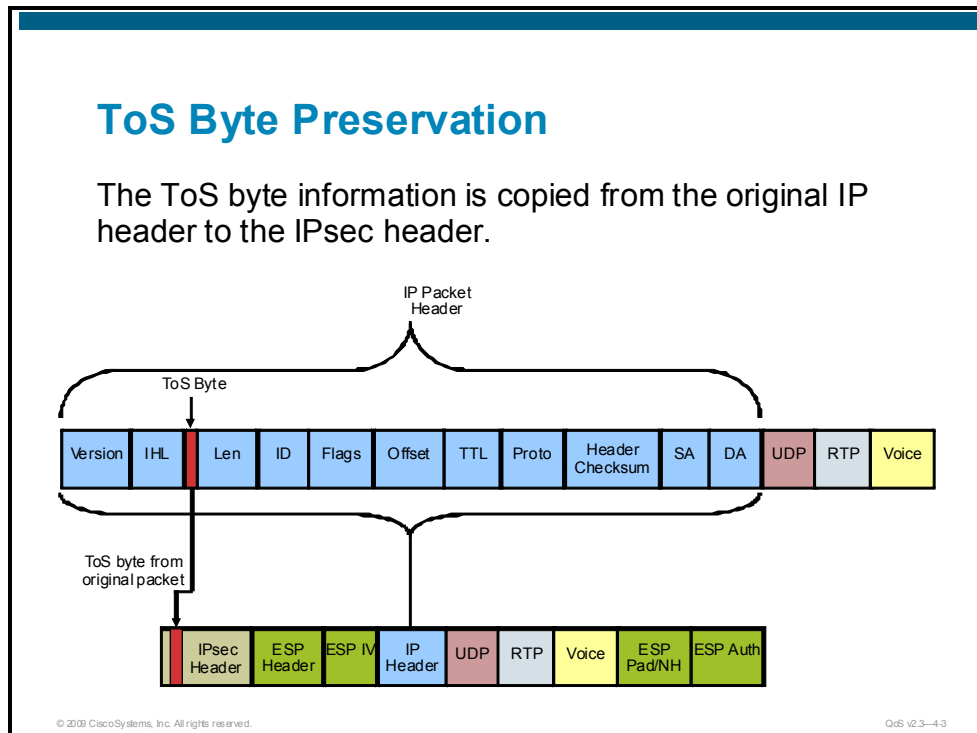
© 2003 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.2

QoS classification is commonly based on the contents of packet headers. However, when an IP packet is encrypted, the IP header becomes unusable by QoS mechanisms that process the packet (post encryption).

ToS Byte Preservation

This topic describes ToS byte preservation.

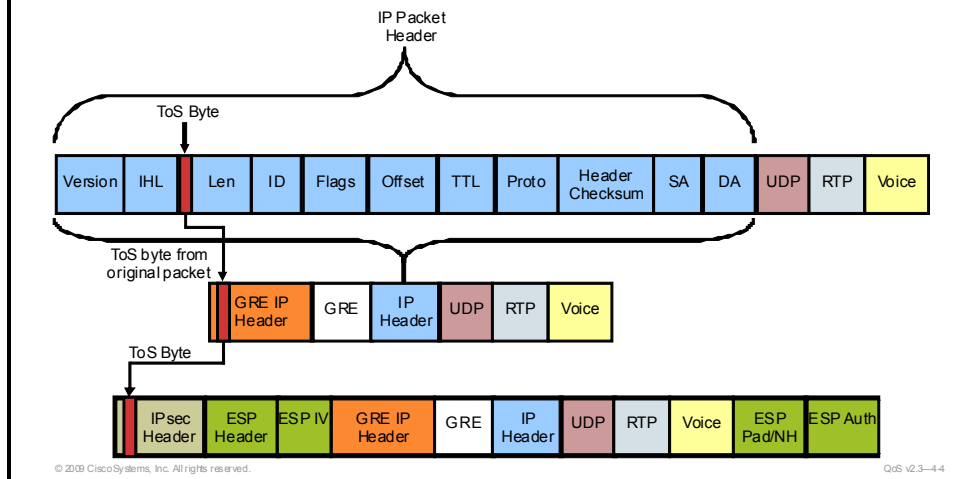


For many QoS designs, classification is performed based on differentiated services code point (DSCP) markings in the ToS byte of the IP packet header. When an IP packet is encrypted through IPsec, the entire packet header (including the ToS byte values) is also encrypted and, thus, unusable by QoS mechanisms that process the packet after encryption.

To compensate for the fact that the ToS byte is unreadable because of encryption, IPsec automatically preserves the ToS byte information of the original IP header by copying it to the IP headers added by the tunneling and encryption process. This process allows the packet to be processed (post-encryption) by QoS mechanisms in the same manner as any other packet.

ToS Byte Preservation (Cont.)

The ToS byte information is copied from the original IP header to the headers added by the tunneling and encryption process.



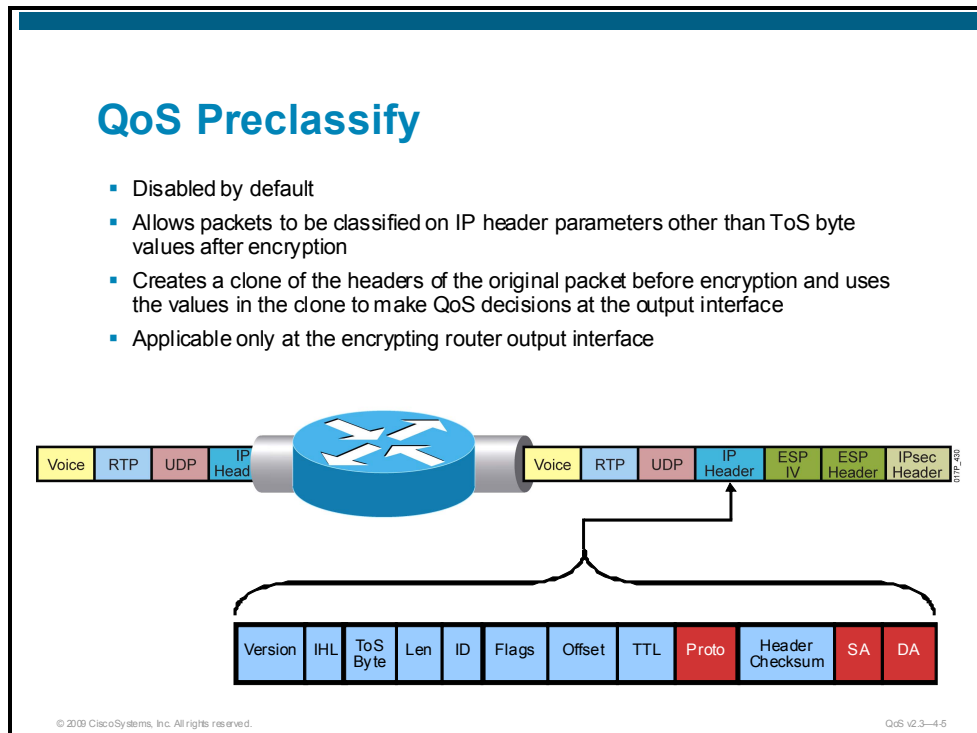
In Cisco IOS Software Release 11.3T and later, the ToS byte information of the original IP header is also automatically copied to the GRE IP header when the router is configured to encapsulate packets using GRE. GRE tunnels based on RFC 1702 allow any protocol to be tunneled in an IP packet. GRE tunneling allows routers between GRE-based tunnel endpoints to adhere to precedence bits, thereby improving the routing of premium service packets. Cisco IOS technologies such as policy routing, weighted fair queuing (WFQ), and weighted random early detection (WRED), can operate on intermediate routers between GRE tunnel endpoints.

GRE tunnels are commonly used to provide dynamic routing resilience over IPsec. Normal IPsec configurations cannot transfer routing protocols, such as Enhanced Interior Gateway Routing Protocol (EIGRP) and Open Shortest Path First (OSPF), or non-IP traffic, such as Internetwork Packet Exchange (IPX) and AppleTalk.

The figure illustrates ToS byte preservation in a scenario where IPsec is being used with GRE. The original IP ToS byte values are copied initially to the IP header added by the GRE encapsulation; the values are then copied again to the IP header added by IPsec encryption. This process allows the packet to be processed (post-encryption) by QoS mechanisms in the same manner as any other packet.

QoS Preclassify

This topic describes the QoS preclassify feature of Cisco IOS Software.



The QoS preclassify feature (also called QoS for VPNs) is often confused with ToS byte preservation. QoS preclassify is a Cisco IOS feature that allows for packets to be classified on header parameters other than ToS byte values after encryption.

Because all original packet header fields are encrypted, including source or destination IP addresses, Layer 4 protocol, and source or destination port addresses, post-encryption QoS mechanisms cannot perform classification against criteria specified within any of these fields. A solution to this constraint is to create a clone of the original packet headers before encryption and then use the values in the clone to make QoS decisions at the output interface.

A key point to remember regarding QoS preclassify is that it is applicable only at the encrypting router output interface. The fields preserved by QoS preclassify are not available to any routers downstream; the clone never leaves the router performing the encryption, thus ensuring the integrity and security of the IPsec VPN tunnel.

QoS preclassify is supported for both GRE and IPsec and is available for these platforms:

- Cisco 7100 Series VPN Routers and Cisco 7200 Series Routers (since Cisco IOS Software Release 12.1(5)T)
- Cisco 2600 and 3600 Series Routers (since Cisco IOS Software Release 12.2(2)T)
- Cisco ASR 1000 Series Routers (since Cisco IOS Software XE Release 2.1)

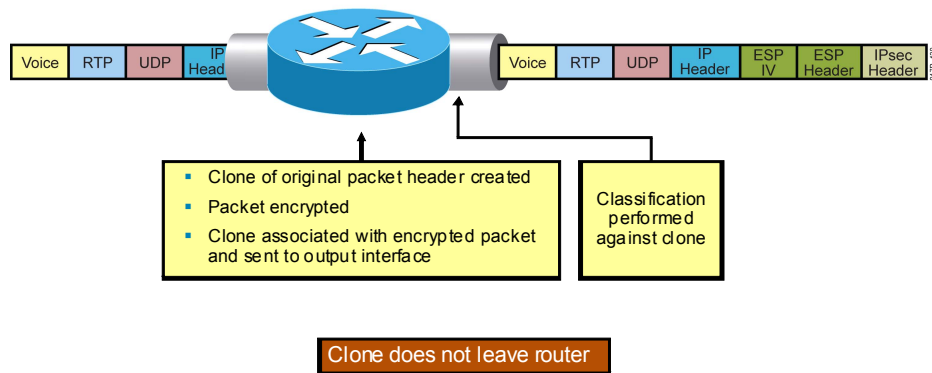
QoS preclassify is supported in all Cisco IOS switching paths and is recommended to be enabled on some platforms even when only the ToS byte values are being used for classification. (QoS preclassify is disabled by default.) Testing has shown that when hardware-based encryption cards are combined with QoS, the Cisco IOS Software implementation of the QoS preclassify feature slightly enhances performance, even when matching only on ToS byte values. Furthermore, enabling QoS preclassify by default eliminates the possibility that its configuration will be overlooked if the QoS policy is later changed to include matching on IP addresses, ports, or protocols.

Design recommendations for the QoS preclassify feature can be summarized as follows:

- Enable QoS preclassify on all branch IPsec VPN routers that support it.
- Enable QoS preclassify on headend IPsec VPN routers only when both the VPN termination and QoS policies reside on the same device.

Note	If a packet is fragmented after encryption, only the first fragment is preclassified. Subsequent fragments might receive different classifications. This behavior is consistent with QoS classification of nontunneled fragments.

QoS Preclassify Feature Operation



QoS preclassify creates a clone of the original packet headers before encryption. The crypto engine encrypts the original packet, and then the clone is associated with the newly encrypted packet and sent to the output interface. At the output interface, QoS decisions based on header criteria can be performed by matching values of the clone. In this way, advanced classification can be administered, even on encrypted packets.

Configuring QoS Preclassify

This topic identifies the Cisco IOS commands required to configure preclassification.

Configuring QoS Preclassify

```
router(config-if)#
```

```
qos pre-classify
```

- Enables the QoS preclassification feature
- Restricted to tunnel interfaces, virtual templates, and crypto maps
- Introduced for Cisco 2600 and 3600 in Cisco IOS Release 12.2(2)T

GRE Tunnels

```
router(config)# interface tunnel0  
router(config-if)# qos pre-classify
```

L2TP Tunnels

```
router(config)# interface virtual-template1  
router(config-if)# qos pre-classify
```

IPSec Tunnels

```
router(config)# crypto map secured-partner  
router(config-crypto-map)# qos pre-classify
```

© 2003 Cisco Systems, Inc. All rights reserved.

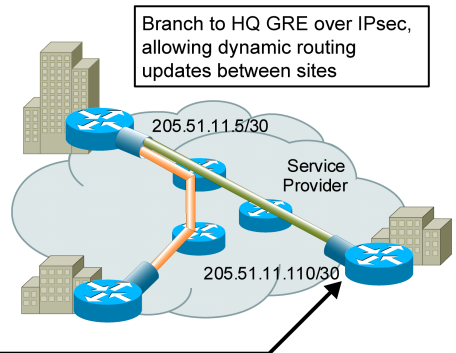
QoS v2.3-4.7

You can use the **qos pre-classify** Cisco IOS command to enable the QoS preclassification feature. Where you apply the command depends upon the type of VPN tunnel that you are using. For GRE tunnels, apply the command to a tunnel interface. For IPsec tunnels, apply the command to a crypto map. When configuring an IPsec encrypted IP GRE tunnel, apply the **qos pre-classify** command to both the tunnel interface and the crypto map. This command can be applied only to a tunnel interface, a crypto map, or a virtual template interface. Virtual template interfaces are used with Layer 2 Tunneling Protocol (L2TP) tunnels; when configuring L2TP tunnels, apply the command to a virtual-template interface.

Note If the classification policy matches on the ToS byte, it is not necessary to use the **qos pre-classify** command.

Configuring QoS Preclassify (Cont.)

```
class-map match-any branch110
match access-group 110
!
policy-map branch-qos
class branch110
bandwidth 128
police 256000
!
interface Tunnel0
ip address 192.168.16.110 255.255.255.0
tunnel source serial0/0
tunnel destination 205.51.11.5
crypto map vpn
qos pre-classify
!
crypto map vpn 10 ipsec-isakmp
set peer 205.51.11.5
set transform-set branch-vpn
match address 110
qos pre-classify
!
interface serial0/0
ip address 205.51.11.110 255.255.255.252
service-policy output branch-qos
crypto map vpn
!
access-list 110 permit gre host
205.51.11.110 host 205.51.11.5
```



The figure shows the successful configuration of the **qos pre-classify** command.

The configuration of the branch router is shown as follows:

- On the S0/0 interface, there is an outgoing service policy that sets the bandwidth of the interface at 128 kb/s and is policed at a rate of 256 kb/s. This policy is applied to any match in the class map branch 110.
- A traffic tunnel has been built on interface S0/0 (whose destination is HQ for this branch IP address 205.51.11.5). It is on this traffic tunnel that QoS preclassification has been configured.

The example configuration also shows that QoS preclassify has been successfully enabled on the crypto map named vpn. This crypto map has also been applied to S0/0. If QoS preclassify is only enabled on the crypto map and not on the tunnel interface, the router will see one flow only, the GRE tunnel (protocol 47).

Monitoring QoS Preclassify

This topic identifies the Cisco IOS commands used to monitor QoS preclassification.

Verifying QoS Preclassify Configuration

```
router>
```

```
show interfaces
```

- Displays traffic seen on a specific interface
- Used to verify that QoS preclassify has been successfully enabled

```
router> show interfaces
Tunnel0 is up, line protocol is up
Hardware is Tunnel
Internet address is 192.168.16.110/24
Tunnel source 205.51.11.110 (Serial0/0), destination 205.51.11.5
Tunnel protocol/transport GRE/IP, key disabled, sequencing disabled
Checksumming of packets disabled, fast tunneling enabled
Last input 00:00:04, output 00:00:04, output hang never
Last clearing of "show interface" counters 00:00:51
Queueing strategy: fifo (QOS pre-classification)
Output queue 0/0, 0 drops; input queue 0/75, 0 drops
```

© 2003 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.0

Use the **show interfaces** command to verify that the QoS preclassify feature has been enabled. Verify by examining the queuing strategy line in the figure:

Queueing strategy: fifo (QOS pre-classification)

Verifying QoS Preclassify Configuration (Cont.)

router>

```
show crypto map [interface interface | tag map-name]
```

- Displays the current crypto map configuration
- Used to verify that QoS preclassify has been successfully enabled on a crypto map

```
router> show crypto map
```

```
Crypto Map "vpn" 10 ipsec-isakmp
Peer = 205.51.11.5
Extended IP access list 110
  access-list 110 permit gre host 205.51.11.110 host 205.51.11.5
Current peer:205.51.11.5
Security association lifetime: 4608000 kilobytes/86400 seconds
PFS (Y/N): N
Transform sets={ branch-vpn, }
QoS pre-classification
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS V2.3-4.10

In the example, the **show crypto map** command has been issued. This command will show current crypto map configuration and also whether the QoS preclassify feature has been successfully enabled on a crypto map.

show crypto map [interface *interface* | tag *map-name*]

Syntax Description

Parameter	Description
interface <i>interface</i>	(Optional) Displays only the crypto map set applied to the specified interface.
tag <i>map-name</i>	(Optional) Displays only the crypto map set with the specified <i>map-name</i> .

Verifying that QoS Preclassify Is Working as Expected

router>

```
show policy-map interface [interface-type interface-number  
| null 0 ] [input | output]
```

- Displays statistics of the policies that are attached to an interface

```
RouterB> show policy-map interface GigabitEthernet0/0  
GigabitEthernet0/0  
  
Service-policy output: branch-qos  
  
Class-map: branch110 (match-all)  
263 packets, 34524 bytes  
30 second offered rate 1000 bps  
Match: access-group 101  
  
Class-map: class-default (match-any)  
3 packets, 228 bytes  
30 second offered rate 0 bps, drop rate 0 bps  
Match: any  
...
```

© 2009 Cisco Systems, Inc. All rights reserved.

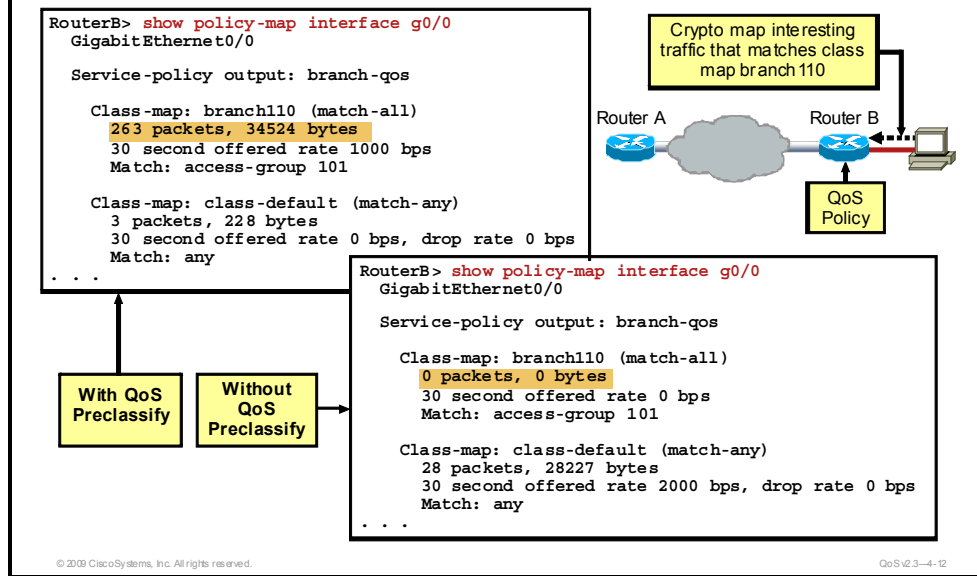
QoS v2.3-4-11

After you have configured a VPN and QoS for your VPN, you can use the **show policy-map interface** command to verify that QoS preclassify is actually working as expected. The **show policy-map interface** command output in the figure shows that class map branch110 has matched 263 packets traversing interface GigabitEthernet0/0, while the default traffic class has matched only 3 packets. If GigabitEthernet0/0 both terminates the VPN and holds the QoS policy, these packet counts are a likely indication that preclassify is working; however, to completely verify the fact, use a procedure such as the following.

Note

For Cisco 3660, 3845, 7200, 7400, 7500, and ASR 1000 Series Routers, the counters displayed after the **show policy-map interface** command is entered are updated only if congestion is present on the interface. For more information on the counters displayed by the **show policy-map interface** command, refer to the *Understanding Packet Counters in show policy-map interface Output* document on <http://www.cisco.com>.

Verifying that QoS Preclassify Is Working as Expected: Example



You can use the **show policy-map interface** command with a procedure such as the following to verify that QoS preclassify is working as expected:

- Step 1** Configure an IPsec VPN between router A and router B. The VPN configuration must meet the following criteria:
 - Telnet traffic must be designated as "interesting traffic" (traffic to be encrypted).
 - The crypto map must include the **qos pre-classify** command.
- Step 2** On router B, configure a QoS policy that contains a single class map. Apply the QoS policy to the VPN output interface of the router. The class map must meet the following criteria:
 - The class map must classify traffic by examining something in the packet header other than ToS byte values.
 - The class map must identify Telnet traffic.
- Step 3** From a host on the router B side of the VPN, generate some Telnet traffic.
- Step 4** Execute the **show policy-map interface** command on router B. If classification is working properly with QoS preclassify configured, the counters for the class map should show packets and bytes as matching the class map.
- Step 5** Remove the **qos pre-classify** statement from the crypto map on router B.
- Step 6** Use the **clear counters** command to clear all interface counters, including the counters for the class map.
- Step 7** Generate Telnet traffic again.
- Step 8** Execute the **show policy-map interface** command on router B. The class map counters should show no traffic matching the class map, thus verifying that the QoS preclassify feature worked as expected when configured.

Summary

This topic summarizes the key points discussed in this lesson.

Summary

- When packets are encapsulated by tunnel or encryption headers, QoS features are unable to examine the original packet headers and correctly classify the packets.
- To compensate for the fact that tunneling or encryption processes render the IP header unreadable by QoS mechanisms, ToS byte preservation automatically copies ToS byte information from the original IP header to the headers added by those processes.
- QoS preclassify is a configurable feature that allows for packets to be classified on header parameters other than ToS byte values after encryption.
- QoS preclassify is enabled by the **qos pre-classify** Cisco IOS command, which can be used on tunnel interfaces, virtual templates, and crypto maps.
- The **show interface** command can be used to verify if QoS preclassify has been enabled.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-13

Configuring QoS Policy Propagation via BGP

Overview

The quality of service (QoS) Policy Propagation via Border Gateway Protocol (BGP) feature, or QPPB, allows packets to be classified based on access lists, BGP community lists, and BGP autonomous system (AS) paths. The supported classification policies include IP precedence setting and the ability to tag the packet with a QoS class identifier internal to the router. After a packet has been classified, you can use other QoS features, such as committed access rate (CAR) and weighted random early detection (WRED), to specify and enforce business policies to fit the business model.

This lesson describes the QPPB classification mechanism. QPPB features covered in this lesson include a review of Cisco Express Forwarding and the tasks and Cisco IOS commands that are required to configure QPPB on Cisco routers.

Objectives

Upon completing this lesson, you will be able to explain how to implement classification and marking in an interdomain network using QPPB. This ability includes being able to meet these objectives:

- Describe the QPPB feature that propagates QoS policy via BGP
- Describe the interaction between QoS and BGP
- Describe Cisco Express Forwarding switching on Cisco IOS platforms
- List the steps required to configure QPPB on Cisco routers
- Identify the Cisco IOS commands required to configure QPPB on Cisco routers

QoS Policy Propagation via BGP

This topic describes the QPPB feature, which propagates QoS policy via BGP.

QoS Policy Propagation on BGP

- QPPB uses **BGP attributes** to advertise CoS to other routers in the network.
- **BGP communities** are usually used to propagate CoS information bound to IP networks.
- Packet classification policy can be propagated on BGP without having to use complex access lists at each of a large number of border (edge) routers.
- A route map is used to translate BGP information (for example, BGP community value) into **IP precedence** or **QoS group**.
- QPPB can only classify and mark inbound packets.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3—4.2

BGP is an interdomain routing protocol that exchanges reachability information with other BGP systems. The QoS policy propagation via the BGP feature allows classifying packets based on access lists, BGP community lists, and BGP AS paths.

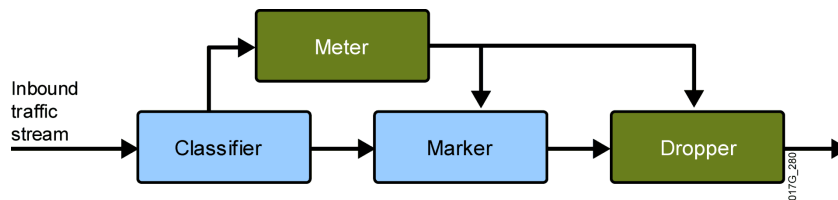
The supported classification policies include IP precedence setting and the ability to tag the packet with a QoS class identifier internal to the router. After a packet has been classified, you can use other QoS features such as policing, WRED, and traffic shaping to specify and enforce business policies to fit the business model.

The QPPB feature has these enhancements:

- **QoS group ID:** An internal QoS group ID can be set that can be used later to perform policing or weighted fair queuing (WFQ) based on the QoS group ID.
- **Source and destination address lookup:** Specify whether the IP precedence level or QoS group ID is obtained from the source (input) address or destination (output) address entry in the route table.

BGP Marking

- Propagate the CoS by encoding it into BGP attributes:
 - BGP communities
 - AS paths
 - IP prefixes
 - Any other BGP attribute
- Translate the selected BGP attribute into either:
 - IP precedence
 - QoS group
- Enable Cisco Express Forwarding and packet marking on interfaces



© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.3

The figure shows three elements used on inbound packets to support QPPB operation.

- Propagate the class of service (CoS) by encoding a value based on the following BGP attributes:
 - BGP communities
 - AS paths
 - IP prefixes
 - Any other BGP attribute
- Translate the selected BGP attribute into either:
 - IP precedence
 - QoS group
- Enable Cisco Express Forwarding and packet marking on interfaces.

QoS and BGP Interaction

This topic describes the interaction between QoS and BGP.

QoS and BGP Interaction

- QoS features work **independently** of BGP routing.
- BGP is used only to **propagate** policies for source or destination IP prefixes through the network.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.4

When using QPPB, the QoS feature works independently from BGP routing. BGP is only used to propagate the QoS policy.

In QPPB configurations, you specify whether to use IP precedence or the QoS group ID obtained from the source (input) address or destination (output) address entry in the routing table. You can specify either the input or output address.

Cisco Express Forwarding

This topic describes Cisco Express Forwarding switching on Cisco IOS platforms.

Cisco Express Forwarding

- The two main components of Cisco Express Forwarding operation:
 - Forwarding Information Base
 - Adjacency tables
- Cisco Express Forwarding was first introduced on the following platforms:
 - Cisco 7x00 Series in Cisco IOS Release 11.1CC
 - All RISC-based platforms in Cisco IOS Release 12.0

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.5

When you initialize the router for Cisco Express Forwarding, two tables are created in the router:

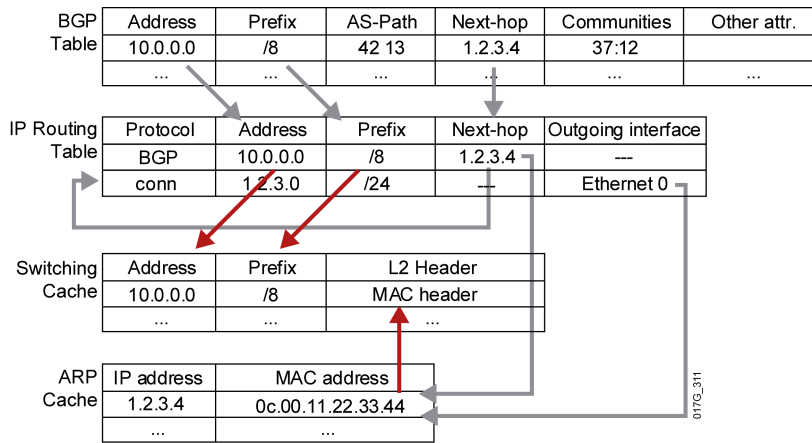
- **Forwarding Information Base (FIB) table:** Lists all paths to all reachable networks, together with the output interface information
- **Adjacency table:** Lists all required next-hops on output interfaces

To enable scalable forwarding, Cisco Express Forwarding builds a forwarding table called the FIB. Contrary to demand-switching methods, the FIB is not a small subset of the routing table. The FIB is a full extract of the routing table, with all the forwarding parameters precalculated at the time of FIB creation, and updated with any topology (routing table) changes.

The second table is the adjacency table, which contains all the Layer 2 next-hops currently being used by the router to forward traffic.

The two tables are interconnected, so that every destination network is linked to its appropriate local next-hop adjacency. Many destinations can be linked to the same next-hop adjacency, removing redundancy and increasing manageability of Cisco Express Forwarding tables. Moreover, a single destination can point to multiple next-hop adjacencies, enabling flexible traffic load-balancing.

Cisco Express Forwarding Review: Standard IP Switching

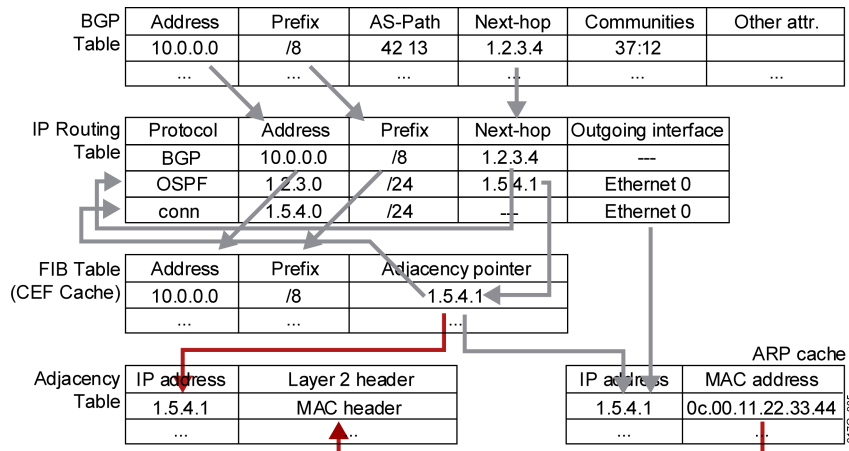


The figure shows a sequence of events when process switching and fast switching for destinations learned via BGP, as follows:

Step 1	When a BGP update is received and processed, an entry is created in the routing table.
Step 2	When the first packet arrives for this destination, the router tries to find the destination in the fast-switching cache. Because the packet is not there, process switching has to switch the packet when the process is run. The process performs a recursive lookup to find the outgoing interface. The process may possibly trigger an Address Resolution Protocol (ARP) request or find the Layer 2 address in the ARP cache. Finally, the process creates an entry in the fast-switching cache.
Step 3	All subsequent packets for the same destination are fast-switched. The switching occurs in the interrupt code (the packet is processed immediately). Fast destination lookup is performed (no recursion).

The encapsulation uses a pregenerated Layer 2 header that contains the destination as well as Layer 2 source (MAC) address. No ARP request or ARP cache lookup is necessary.

Cisco Express Forwarding Review: Cisco Express Forwarding Switching



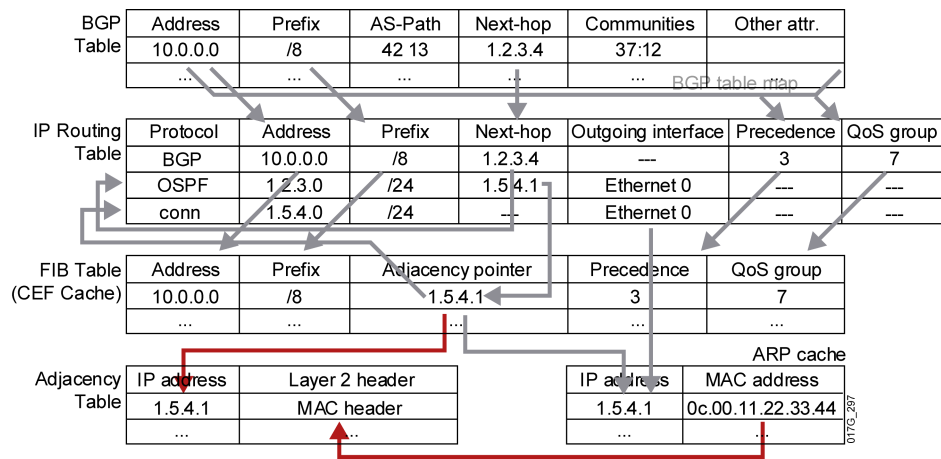
The generation of entries in the FIB table is not packet-triggered, but change-triggered. When something changes in the IP routing table, the change is also reflected in the FIB table.

Because the FIB contains the complete IP switching table, the router can make definitive decisions based on the FIB. Whenever a router receives a packet that should be Cisco Express Forwarding-switched, but the destination is not in the FIB, the packet is dropped.

The FIB table is also different from other fast-switching caches in that it does not contain information about the outgoing interface and the corresponding Layer 2 header. That information is stored in a separate table, the adjacency table. The adjacency table is more or less a copy of the ARP cache, but instead of holding only the destination MAC address, the adjacency table holds the Layer 2 header (source and destination MAC address).

The figure illustrates how the Cisco Express Forwarding switching entries are created. When a route is added or changed in the main routing table (for example, learned via BGP), a new FIB entry is created, and the next hop is calculated via recursive lookups to the routing table (if necessary). The FIB entry is then linked to the next-hop adjacency entry, which provides the necessary Layer 2 information used to forward the packet on the output medium.

Cisco Express Forwarding Switching with QoS Packet Marking



In the figure, the tables are again displayed, with the difference that BGP communities being translated to IP precedence and QoS group are also inserted into the FIB table.

QPPB Configuration Tasks

This topic describes how to configure QPPB on Cisco routers.

QPPB Configuration Tasks

1. Create a route map to set IP precedence or QoS group.
2. Apply the route map to BGP routes transferred to main IP routing table.
3. Enable per-interface packet marking.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.9

When setting up a BGP policy, there are three required steps to consider:

- Step 1** Create a route map to set IP precedence or QoS group. Use the **route-map** command to accomplish this task, as follows:

```
route-map <route-map name> permit 10
  match community <community-list>
  set ip precedence <IP precedence value>
  set ip qos-group <qos-group #>
```

- Step 2** Apply the route map to BGP routes that are in the BGP table. Use the **table-map** command to accomplish this task, as follows:

```
router bgp <as #>
  table-map <route-map name>
```

- Step 3** Enable the required interface for packet marking. Use the **bgp-policy** command to accomplish this task, as follows:

```
interface X
  bgp-policy <source | destination> ip-prec-map
```

Configuring QPPB

This topic describes the Cisco IOS commands that are required to configure QPPB.

Setting IP Precedence or QoS Group in the IP Routing Table

```
router(config)#
```

```
route-map name permit seq
match as-path path-list-number
match ip address access-list-number
match community community-list
set ip precedence precedence
set ip qos-group group
```

- Defines a route map to set IP precedence or QoS group
- Specifies IP precedence and QoS group values in the routing table/FIB table entry

```
router(config-router)#
```

```
table-map route-map-name
```

- Specifies the route map used to set additional routing table attributes

© 2009 Cisco Systems, Inc. All rights reserved.QoS v2.3-4-10

Use the **route-map** command to define a route map to match based on a BGP community list, BGP AS path, or access list, and to set the IP precedence or QoS group. To set the IP precedence value (and an optional IP number or IP name) in the IP header, use the **set IP precedence** route-map configuration command. To leave the precedence value unchanged, use the **no** form of this command.

The syntax for the **set ip precedence** command is as follows:

```
set ip precedence [precedence | name]
```

Syntax Description

Parameter	Description
<i>precedence name</i>	A number or name that sets the precedence bits in the IP header. The values for the <i>precedence</i> argument and the corresponding <i>name</i> argument are listed in the table from least to most important.

The table lists the values for the *precedence* argument and the corresponding *name* argument for precedence values in the IP header. They are listed from least to most important.

Precedence	Name
0	routine
1	priority
2	immediate
3	flash
4	flash-override
5	critical
6	Internet
7	network

To set a group ID that can be used later to classify packets, use the **set qos-group** QoS policy map configuration command. To remove the group ID, use the **no** form of this command.

set qos-group *group-id*

Syntax Description

Parameter	Description
<i>group-id</i>	Group ID number in the range from 0 to 99

Note To display QoS group information, use the **show ip cef** command.

Use the BGP **table-map** command to apply the route map to the BGP routing process. This will populate the corresponding BGP routes in the IP routing table and FIB with the CoS (IP precedence and/or QoS group) information. To modify metric and tag values when the IP routing table is updated with BGP learned routes, use the **table-map** command in address family or router configuration mode. To disable this function, use the **no** form of the command.

The syntax for the **table-map** command is as follows:

`table-map map-name`

Syntax Description

Parameter	Description
<i>map-name</i>	Route map name, from the route-map command

The syntax shown in the figure for the **match as-path** and **table-map** commands is the same as the syntax for the **match as-path** and **table-map** commands in Cisco IOS Software Release 12.4.

The syntax shown in the figure for the **route-map**, **match ip address**, **match community**, **set ip precedence**, and **set ip qos-group** commands is compatible with Cisco IOS Software Release 12.4.

Enable Per-Interface Packet Marking

```
router(config-if)#
```

```
bgp-policy {source | destination} ip-prec-map
```

- Marks packets using the IP precedence based on the packet source address or destination address.
- If both source and destination are specified on an interface, the software lookup for the destination address occurs last and the packet is re-marked based on the destination address.

```
router(config-if)#
```

```
bgp-policy {source | destination} ip-qos-map
```

- Marks packets using the QoS group ID based on the packet source address or destination address.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.11

After the CoS information (IP precedence or QoS group) in the IP routing table and the FIB table, you can configure Cisco Express Forwarding-based markings on the input interfaces by using the **bgp-policy** interface configuration command.

Using the **bgp-policy** interface configuration command, you can furnish the Cisco Express Forwarding-based markings based on the source or destination address of an incoming packet. Use the source option to mark packets sourced from a customer. Use the destination option to mark packets destined to a customer.

You can mark the packets with the IP precedence or QoS group value from the FIB table. Use the **ip-prec-map** option to mark the packets with IP precedence, and use the **ip-qos-map** option to mark the packets with a QoS group.

The syntax for the **bgp-policy** command is as follows:

```
bgp-policy {source | destination} {ip-prec-map | ip-qos-map}
```

Syntax Description

Parameter	Description
source	The IP precedence bit or QoS group ID from the source address entry in the route table
destination	The IP precedence bit or QoS group ID from the destination address entry in the route table
ip-prec-map	QoS policy based on the IP precedence
ip-qos-map	The QoS policy based on the QoS group ID

The syntax shown in the figure for the **bgp-policy** command is compatible with Cisco IOS Software Release 12.4.

Note If you specify both **source** and **destination** on the interface, the software looks up the source address in the routing table and classifies the packet based on the source address first; then the software looks up the destination address in the routing table and reclassifies it based on the destination address.

Example: Configuration

```
router bgp 30
  table-map precedence-map
  neighbor 20.20.20.1 remote-as 10
  neighbor 20.20.20.1 send-community
  !
  ip bgp-community new-format
  !
  ! Match community 1 and set the IP precedence to priority and
  set the QoS group to 1
  route-map precedence-map permit 10
  match community 1
  set ip precedence priority
  set ip qos-group 1
  !
  ! Match community 2 and set the IP precedence to immediate
  route-map precedence-map permit 20
  match community 2
  set ip precedence immediate
  !
  ip community-list 1 permit 60:1
  ip community-list 2 permit 60:2
  !

interface HSSI 5/0
  no ip address
  encapsulation frame-relay
  !
interface HSSI 5/0/0.1 point-to-point
  ip address 20.20.20.1 255.255.255.0
  bgp-policy source ip-prec-map
  no ip mroute-cache
  frame-relay interface-dlci 20 IETF
```

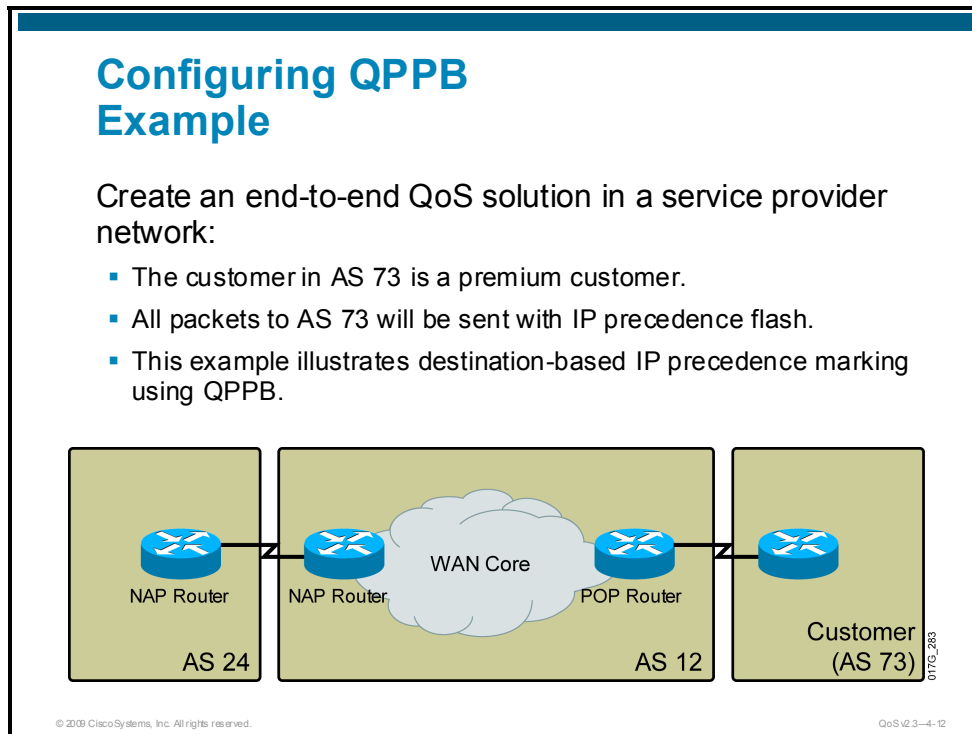
In this example, the community attribute is being matched and then the action is taken on those attributes. If the community is 60:1 (IP community list 1), its IP precedence will be set to priority and the QoS group will be set to 1, as specified in the route map precedence map.

If the community attribute is 60:2 (IP community list 2), its IP precedence will be set to immediate.

The policy is then applied to the interface HSSI 5/0/0/0.1, using the **bgp-policy source** command. The **ip-prec-map** keyword indicates that the QoS policy is based on IP precedence.

Example: Configuring QPPB

The figure shows an example of configuring QPPB.

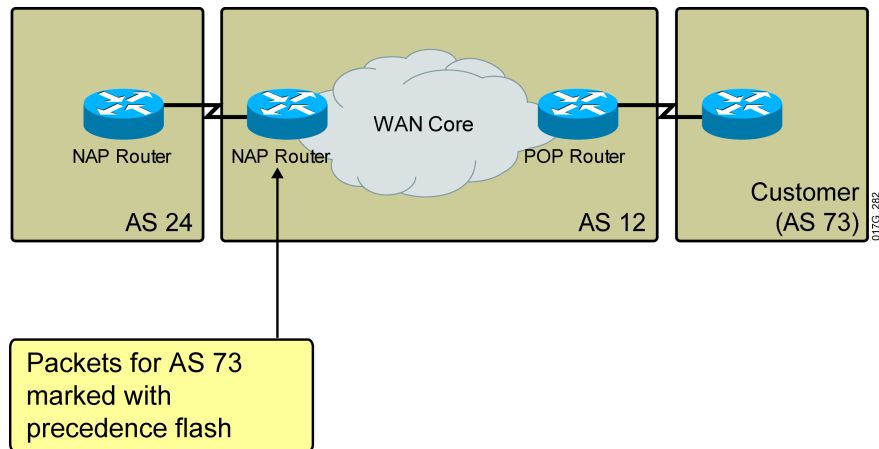


In a service-provider network, an end-to-end QoS solution must be created.

The requirements for the example are as follows:

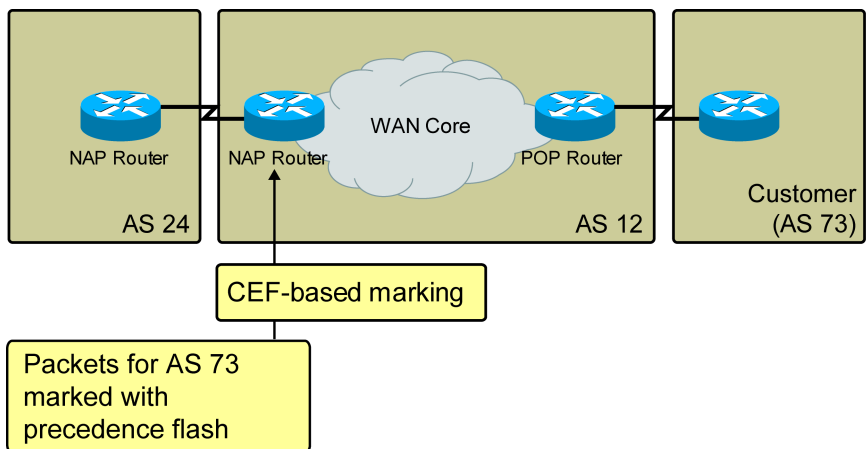
- The customer in AS 73 is a *premium* customer.
- All packets to AS 73 are sent with IP precedence *flash*.

Step 1: Distribute QoS Functions



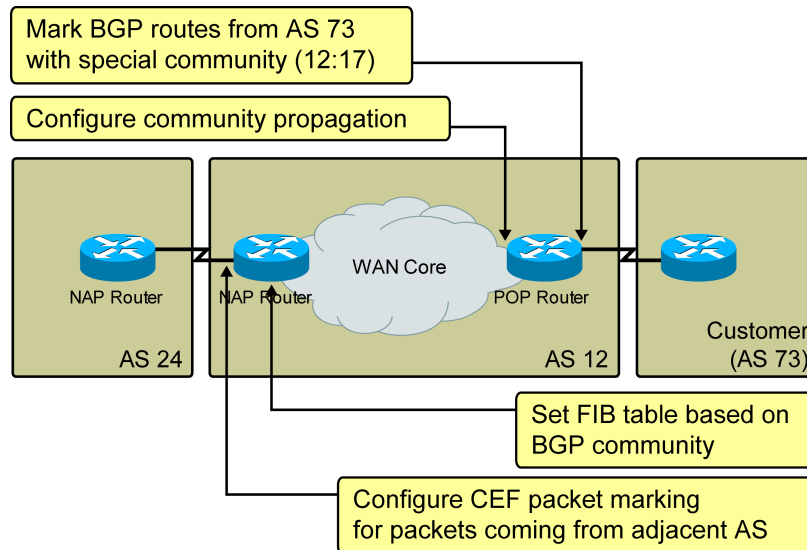
Because you are going to create an end-to-end QoS solution, this figure illustrates the first step: Routes that are received from AS 24 and destined for AS 73 will have IP precedence set to *flash* on the network access point (NAP) router (in AS 12).

Step 2: Select QoS Mechanisms



The figure shows the second requirement: Enable Cisco Express Forwarding-based marking on the NAP router serial interface connecting to AS 24.

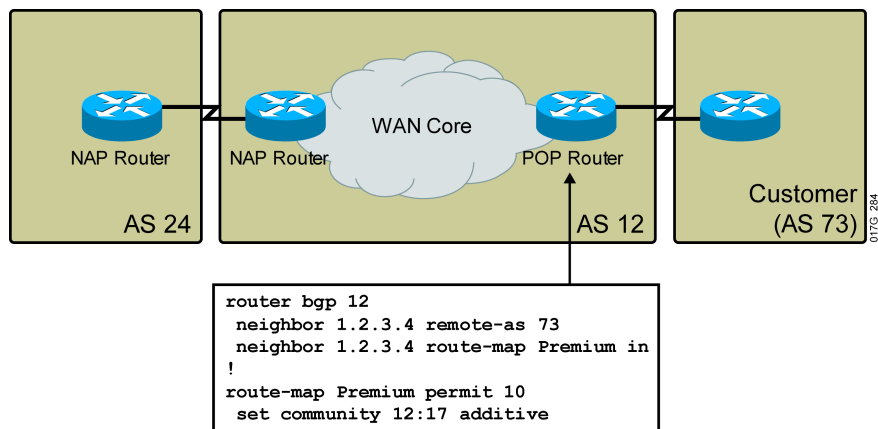
Step 3: Design Individual QoS Mechanisms



This figure shows the third requirement, consisting of the following:

- BGP routes that are received into AS 12 from AS 73 will be marked with a community value of 12:17 on the point of presence (POP) router.
- Community propagation will have to be configured on the POP router so that the community value of 12:17, set on the POP router, will be propagated to the NAP router.
- All the BGP routes with a community of 12:17 in the IP routing table and the FIB table on the AS 12 NAP routers will contain the IP precedence flash.

Mark Routes Coming from AS 73

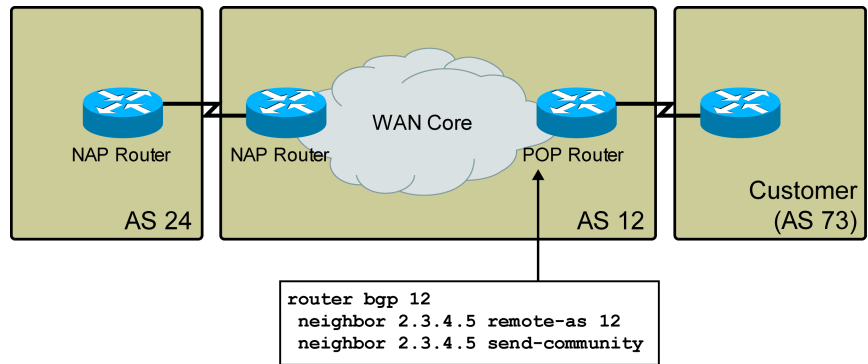


The figure shows the configuration that is necessary to meet the requirement that routes coming from AS 73 will be marked with the special community value of 12:17.

Use this configuration on the POP router to mark BGP routes from AS 73 with the community value 12:17:

```
router bgp 12
neighbor 1.2.3.4 remote-as 73
neighbor 1.2.3.4 route-map Premium in
!
route-map Premium permit 10
set community 12:17 additive
```

Configure Community Propagation

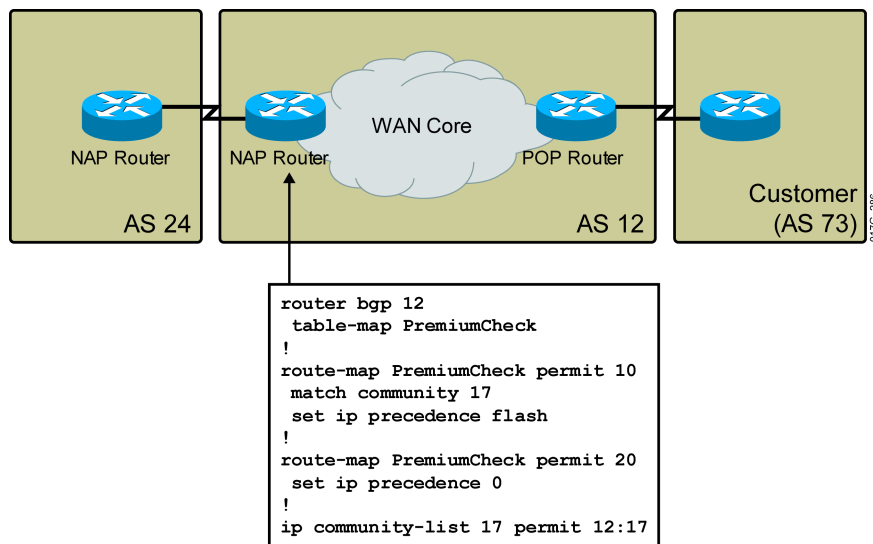


The figure shows the configuration that is necessary to propagate the special community value, 12:17, which has been added on the POP router and will be seen on the NAP router.

The following configuration is used on the POP router to mark to propagate the community value (12:17) to the NAP router:

```
router bgp 12
neighbor 2.3.4.5 remote-as 12
neighbor 2.3.4.5 send-community
```

Set FIB Table Based on BGP Community



The NAP router in AS 12 uses a route map to translate BGP community values into appropriate IP precedence values. This figure illustrates how all BGP routes carrying BGP community 12:17 are tagged with IP precedence flash in the routing table and the FIB table. All other BGP routes are tagged with IP precedence 0.

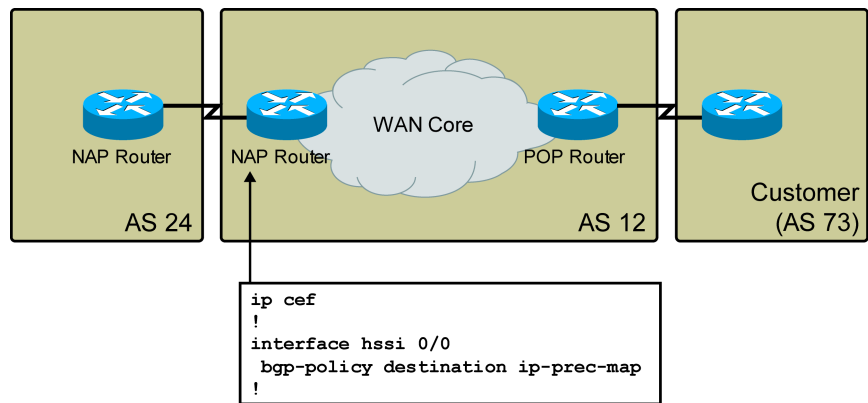
The following configuration is used on the NAP router to set or change the IP precedence of those BGP routes that match the community value (12:17):

```

router bgp 12
 table-map PremiumCheck
 !
 route-map PremiumCheck permit 10
  match community 17
  set ip precedence flash
 !
 route-map PremiumCheck permit 20
  set ip precedence 0
 !
 ip community-list 17 permit 12:17
  
```

The configuration shows that if the route map PremiumCheck matches the community attribute of 12:17, the corresponding packet will have its IP precedence changed to flash, as required in the example.

Configure Cisco Express Forwarding Packet Marking



The last configuration step is to enable Cisco Express Forwarding-based marking on the NAP router in AS 12. This example requires that all packets going to (destination-based marking) the customer network (AS 73) be marked with IP precedence flash.

In this case, the AS 12 NAP router HSSI 0/0 interface connects to AS 24. Therefore, the **bgp-policy destination ip-prec-map** command is configured under the HSSI 0/0 interface to enable destination-based Cisco Express Forwarding-based marking. All packets from AS 24 destined to customer AS 73 will be marked with IP precedence flash.

QPPB marking is only available in combination with Cisco Express Forwarding switching. The global **ip cef** command enables Cisco Express Forwarding switching on all interfaces that support Cisco Express Forwarding.

The following configuration is used on the NAP router to configure Cisco Express Forwarding packet marking:

```
ip cef
!
interface hssi 0/0
  bgp-policy destination ip-prec-map
```

Summary

This topic summarizes the key points discussed in this lesson.

Summary

- The QoS policy propagation via the QPPB BGP feature allows classifying packets based on access lists, BGP community lists, and BGP AS paths.
- When using QPPB, QoS works independently from BGP routing.
- Cisco Express Forwarding switching with QoS **packet marking** will populate the FIB table with IP precedence and QoS group values.
- Route maps are used to set IP precedence and QoS group ID.
- The **bgp-policy** Cisco IOS command (interface level) is used to mark the QoS policy via BGP.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS 4.3-4.3D

Configuring LAN Classification and Marking

Overview

A switch may operate at wire speeds, but because most switches have been inherently designed to accommodate oversubscription—such as having more access ports than uplink ports—the switch may congest. During times of congestion, if the congestion management features are not in place, packets will be dropped. When packets are dropped, retransmissions occur. When retransmissions occur, the network load can increase. In networks that are already congested, an increase in network load can add to existing performance issues and potentially further degrade performance. With converging networks, congestion management is even more critical. Latency-sensitive traffic, such as voice and video, can be severely impacted if delays are incurred. Simply adding more buffers to a switch will not necessarily alleviate congestion problems. Latency-sensitive traffic must be switched as fast as possible. First, this important traffic must be identified through classification techniques, and then buffer management techniques must be implemented to prevent the higher priority traffic from being dropped during congestion.

This lesson describes classification and marking as it is implemented on Cisco Catalyst switches. Topics covered include LAN classification and marking options and platforms, and configuring and monitoring LAN-based classification and marking.

Objectives

Upon completing this lesson, you will be able to describe LAN-based methods for implementing classification and marking. This ability includes being able to meet these objectives:

- Describe LAN-based classification and marking using a Layer 2 Catalyst workgroup switch
- Describe QoS trust boundaries and explain their significance in LAN-based classification and marking
- Describe several of the Catalyst switches and identify the different classification and marking options available on Cisco Layer 2 and Layer 3 switching platforms

- Identify the Cisco IOS commands required to configure LAN-based classification and marking
- Identify some of the Cisco IOS commands required to monitor LAN-based classification and marking

LAN-Based Classification and Marking

This topic describes LAN-based classification and marking using a Layer 2 Catalyst workgroup switch.

LAN-Based Classification and Marking

- Classification and marking should typically be performed as close to the source of the traffic as possible.
- Defining trust boundaries is important when performing classification and marking in the LAN.
- For QoS marking transparency, mapping between Layer 2 and Layer 3 classification schemes must be accomplished.
- Cisco Catalyst switches have classification and marking capabilities and are ideal locations for performing these critical QoS functions.
- Classification and marking mechanisms of workgroup switches are based on DSCP and CoS, but compatibility with IP precedence can be achieved because DiffServ is backwards-compatible.
- Only ports that have been configured as ISL or 802.1Q trunks can carry Layer 2 CoS values.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.2

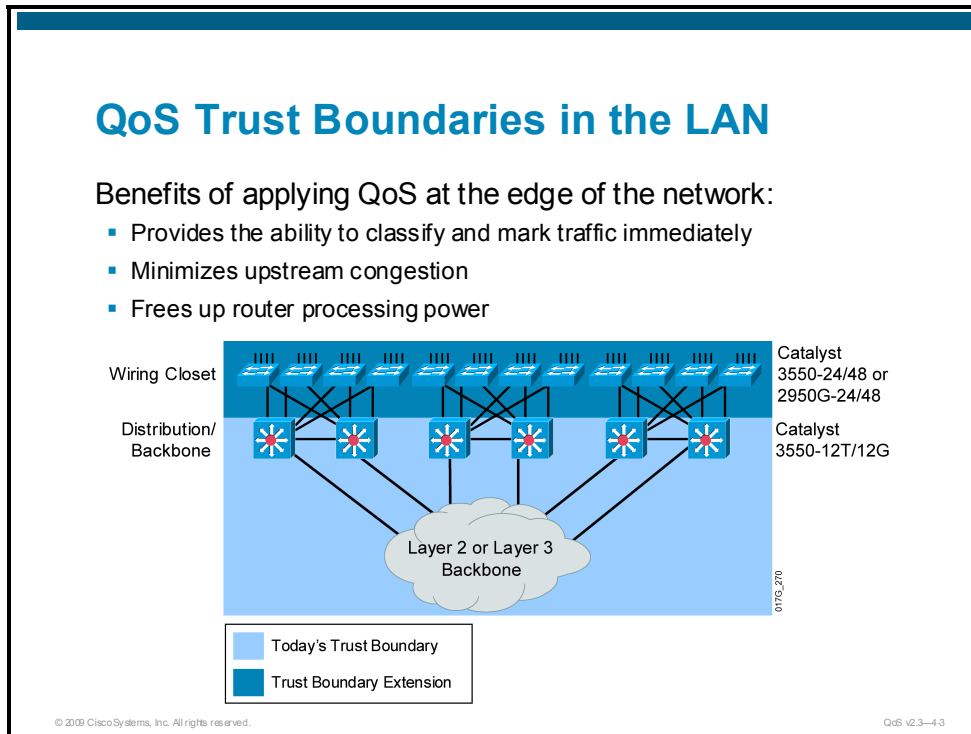
The Cisco Catalyst line of multilayer switches includes the capability to provide quality of service (QoS) at Layer 2 or Layer 3, depending on the switch type. At Layer 2, the frame uses class of service (CoS) in 802.1p and Inter-Switch Link (ISL). CoS uses three bits, similar to IP precedence, and maps well from Layer 2 to Layer 3, and vice versa.

The switches have the capability to differentiate frames based on CoS settings. If multiple queues are present, frames can be placed in different queues and serviced via weighted round robin (WRR) or shaped round robin (SRR) scheduling algorithms.

Classification is only performed on a Cisco Catalyst switch if QoS has been globally enabled on the switch.

QoS Trust Boundaries

This topic describes QoS trust boundaries and their significance in LAN-based classification and marking.



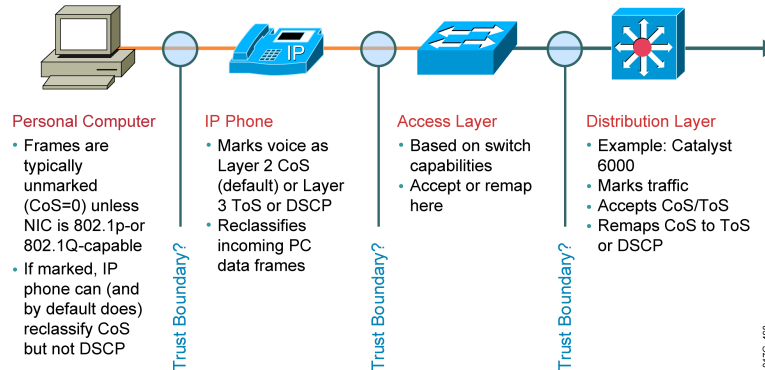
You should apply QoS as close to the source of the traffic as possible.

These are some of the benefits of applying QoS at the edge (or close to the source of the traffic):

- Provides the ability to classify and mark traffic immediately
- Reduces CPU use of upstream devices, thus reducing the possibility that priority traffic, such as voice, is delayed at some point further in the network
- Frees up router processing power, as Catalyst switches perform QoS operations in hardware (with no incremental CPU load)

QoS Trust Boundary in the LAN: Classify and Mark Where?

- Cisco QoS model assumes that the CoS carried in a frame may or may not be trusted by the network device.
- Classification should be done as close to the edge as possible.
- End hosts such as user PCs generally cannot be trusted to correctly tag a packet priority.



Classification should take place at the network edge, typically in the wiring closet or within video endpoints or IP phones themselves.

The figure demonstrates this with an IP telephony example. Packets can be marked as important by using Layer 2 CoS settings in the user priority bits of the 802.1p portion of the 802.1p or 802.1Q fields, or the IP precedence and differentiated services code point (DSCP) bits in the type of service (ToS) or Differentiated Services (DiffServ) field in the IPv4 header. Cisco IP phones can mark voice packets as high priority using CoS and DSCP. By default, the IP phone sends 802.1p-tagged packets with the CoS set to a value of 5 and the DSCP set to Expedited Forwarding (46).

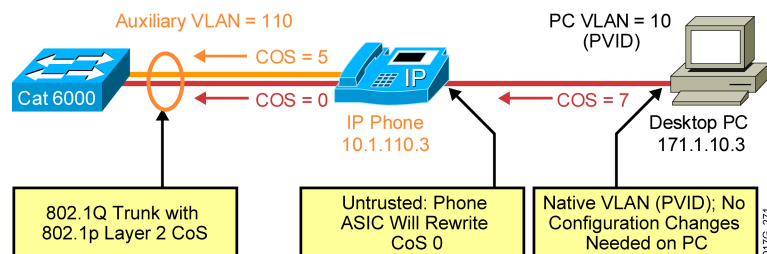
In a Cisco IP telephony environment, PCs are placed in a native VLAN, meaning that their Ethernet packets are untagged. This means that (properly formatted) Ethernet frames originating from a PC will not have an 802.1p field and thus no provision to set CoS values. Additionally, by default, DSCP values of packets originating from PCs are set to 0.

A special case exists when the networking stack in the PC has been modified to send packets with a CoS or DSCP value other than zero. Typically, this does not happen, but even if the PC sends tagged frames with a specific CoS value, Cisco IP phones can zero out the CoS value before sending the frames to the switch. This is the default behavior. Voice frames coming from the IP phone have a CoS of 5 and data frames coming from the PC have a CoS of 0. When the switch receives these frames, it can take into account these values for further processing based on its capabilities.

Some switches also use queues to buffer incoming frames before sending the frames to the switching engine. (It is important to remember that input queuing, such as output queuing, comes into play only when there is congestion.) The switch uses the CoS values to put the frames in appropriate queues. The switch can employ congestion management mechanisms, such as WRR or SRR, as well as congestion avoidance mechanisms, such as weighted random early detection (WRED), or weighted tail drop (WTD) on either input or output queues.

Connecting the IP Phone

- 802.1Q trunking between the switch and IP phone for multiple VLAN support (separation of voice and data traffic) is preferred.
- The 802.1Q header contains the VLAN information and the CoS 3-bit field, which determines the priority of the packet.
- For most Cisco IP phone configurations, traffic sent from the IP phone to the switch is trusted to ensure that voice traffic is properly prioritized over other types of traffic in the network.
- The trusted boundary feature uses Cisco Discovery Protocol to detect an IP phone and otherwise disables the trusted setting on the switch port to prevent misuse of a high-priority queue.



© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.6

In a typical network, a Cisco IP phone is connected to a switch port as shown in the figure. Traffic sent from the telephone to the switch is typically marked with a tag that uses the 802.1Q header. The header contains the VLAN information and the CoS three-bit field, which determines the priority of the packet. For most Cisco IP phone configurations, the traffic sent from the telephone to the switch is trusted to ensure that voice traffic is properly prioritized over other types of traffic in the network. By using the **mls qos trust device cisco-phone** and the **mls qos trust cos** interface configuration commands, the switch port to which the telephone is connected can be configured to trust the CoS labels of all traffic received on that port.

LAN Classification and Marking Platforms

This topic describes several of the Cisco Catalyst switches, identifying the different classification and marking options available on Layer 2 and Layer 3 switching platforms.

Classification and Marking on Catalyst Switches				
	6500 (PFC)	4500 (Sup II plus, III, IV, V)	3750	3550 , 3560 (2970)
Trust Capabilities	CoS DSCP IP precedence (Module-dependent) Extend trust to IP phone	CoS DSCP --- (Module-dependent) Extend trust to IP phone	CoS DSCP IP Precedence IP Phone Extend trust to IP phone	CoS DSCP IP Precedence IP Phone Extend trust to IP phone
CoS to DSCP and DSCP to CoS Mapping Tables	Yes	Yes	Yes	Yes
IPP to DSCP Mapping Table	Yes	Yes	Yes	Yes
DSCP Options (pass-thru, mutation)	Yes	Yes (no mutation)	Yes	Yes
ACL	Yes	Yes	Yes	Yes
Class-Based Markings	Yes	Yes	Yes	Yes

© 2009 Cisco Systems, Inc. All rights reserved. QoS v2.3-4.6

Catalyst 6500

Classification can be performed only on the Catalyst 6500 Switch when QoS has been globally enabled. By default, QoS is globally disabled and classification does not occur.

With a policy feature card (PFC), QoS can classify traffic that is addressed to a specified MAC address, or VLAN pairs to be marked with a configured CoS value. Classification can also be performed against Layer 3 and Layer 4 values. Marking may be by Layer 2 CoS or Layer 3 IP precedence or DSCP values.

QoS schedules traffic through the transmit queues based on CoS values and uses CoS-value-based transmit-queue drop thresholds to avoid congestion in traffic that is transmitted from Ethernet-based ports. The implementation of scheduling and congestion avoidance is line card hardware-dependent, and with each specific line card family different queue capabilities exist.

Queues are defined in terms of number of queues, the type of queue, and the number of drop thresholds per queue. Here are three examples:

- **2q2t:** Indicates two standard queues, each with two configurable tail-drop thresholds
- **1p2q2t:** Indicates one strict-priority queue and two standard queues, each with two configurable WRED drop thresholds
- **1p3q1t:** Indicates one strict-priority queue and three standard queues, each with one configurable WRED drop threshold (On 1p3q1t ports, each standard queue also has one nonconfigurable tail-drop threshold.)

With 1p3q1t, the three standard transmit queues each have one WRED drop threshold and one nonconfigurable tail-drop threshold, and the frames, by default, are handled as follows:

- Frames with CoS 5 go to the strict-priority transmit queue (queue 4), where the switch drops frames only when the buffer is 100 percent full.
- Frames with CoS 0 and 1 go to the low-priority standard transmit queue (queue 1).
- Frames with CoS 2, 3, or 4 go to the medium-priority standard transmit queue (queue 2).
- Frames with CoS 6 or 7 go to the high-priority standard transmit queue (queue 3).

Catalyst 4500

Classification on the Catalyst 4500 Switch can be performed only if QoS is globally enabled on the switch. By default, QoS is globally disabled and classification does not occur.

You must specify which fields in the frame or packet will be used to classify incoming traffic. For IP traffic, you have these classification options:

- Trust the IP DSCP in the incoming packet (configure the port to trust DSCP), and assign the same DSCP to the packet for internal use.
- Trust the CoS value (if present) in the incoming packet, and generate the DSCP by using the CoS-to-DSCP map.
- Perform the classification based on a configured IP standard or extended access control list (ACL), which examines various fields in the IP header. If no ACL is configured, the packet is assigned the default DSCP based on the trust state of the ingress port; otherwise, the policy map specifies the DSCP to assign to the incoming frame.

You can classify a packet for QoS using multiple match criteria, and the classification can specify whether the packet should match all of the specified match criteria or at least one of the match criterion. To define a QoS classifier, provide the match criteria using the “match” statements in a class map. In the “match” statements, specify the fields in the packet to match on, or use IP standard or IP extended ACLs.

During QoS processing of a frame, the switch represents its priority with an internal DSCP value:

- During classification, QoS uses configurable mapping tables to derive the internal DSCP from received CoS. These maps include the CoS-to-DSCP map.
- During policing, QoS can assign another DSCP value (if the packet is out of profile and the policer specifies a marked-down DSCP value).
- Before the traffic reaches the scheduling stage, QoS uses the internal DSCP to select one of the four egress queues for output processing. You can configure the DSCP-to-egress queue mapping by using the **qos map dscp to tx-queue** command.

The CoS-to-DSCP and DSCP-to-CoS maps have default values that might or might not be appropriate for the network.

Each physical port has four transmit queues (egress queues). Each packet that needs to be transmitted is attached to one of the transmit queues. The transmit queues are then serviced based on the transmit queue scheduling algorithm.

After the final transmit DSCP is computed (including any markdown of DSCP), the transmit-DSCP-to-transmit-queue mapping configuration determines the transmit queue. The packet is placed in the transmit queue of the transmit port, determined from the transmit DSCP. Use the **qos map dscp to tx-queue** command to configure the transmit DSCP to transmit queue mapping.

You can configure the transmit queue 3 on each port as the priority queue using the **priority high tx-queue** configuration command in the interface configuration mode. When transmit queue 3 is configured with higher priority, packets in transmit queue 3 are scheduled ahead of packets in other queues.

Catalyst 3750

The Catalyst 3750 Switch supports Layer 3 routing and may be found in either the access layer or the distribution layer. QoS is globally disabled by default on the Catalyst 3750. While QoS is disabled, all frames and packets are passed through the switch unaltered (which is equivalent to a trust CoS and trust DSCP state on all ports). When QoS is globally enabled, however, all DSCP and CoS values are (by default) set to 0 (which is equivalent to an untrusted state on all ports). QoS must be enabled globally for configured policies to become effective.

Generally, the Catalyst 3750 is relatively compatible in QoS features and syntax with the Catalyst 3550, except with respect to queuing and dropping.

As in the Catalyst 6500, queues are defined in terms of number of queues, the type of queue, and the number of drop thresholds per queue. The Catalyst 3750 supports four egress queues, which can be configured on a per-interface basis to operate in either 4q3t or 1p3q3t modes. Additionally, the Catalyst 3750 supports two queue sets, allowing certain interfaces to be configured in one manner and others to be configured in a different manner. For example, some interfaces may be assigned to queue set 1 operating in 4q3t mode, while others may be assigned to queue set 2 operating in 1p3q3t mode.

However, unlike the Catalyst 2950 and 3550, the Catalyst 3750 has queue 1 (not queue 4) as the optional priority queue. In a converged campus environment, it is recommended to enable the priority queue via the **priority-queue out** interface command.

Note	The Catalyst 3750 also supports two configurable ingress queues (normal and expedite). Ingress scheduling, however, is rarely, if ever, required, because it only becomes enabled if the combined input rates from any or all switch ports exceed the capacity of the switch fabric. Such cases are extremely difficult to achieve, even in controlled lab environments. In the extreme case in which such a scenario develops in a production environment, the default settings of the ingress queues are acceptable to maintain VoIP quality and network availability.

The three remaining egress queues on the Cisco Catalyst 3750 are scheduled by SRR, which can be configured to operate in shaped mode or in shared mode.

The Catalyst 3750 supports three WTD thresholds per queue. Two of these thresholds are configurable (explicit); the third is non-configurable (implicit), because it is set to the queue-full state (100%). These thresholds can be defined with the **mls qos queue-set output threshold** global command.

Once the queues and thresholds have been defined, traffic can be assigned to queues and thresholds either by CoS values or DSCP values, using the **mls qos srr-queue output cos-map queue** and **mls qos srr-queue output dscp-map queue** global commands, respectively.

Catalyst 3550

Each Gigabit-capable Ethernet port has four egress queues, one of which can be the egress expedite or priority queue. If the expedite (priority) queue is enabled, WRR services it until the queue is empty before servicing the other three queues. Ingress frame or packet classification options include the following:

■ Non-IP traffic:

- Use the port default. If the frame does not contain a CoS value, the switch assigns the default port CoS value to the incoming frame. Then the switch uses the configurable CoS-to-DSCP map to generate the internal DSCP value.
- Trust the CoS value in the incoming frame (configure the port to trust CoS). Then the switch uses the configurable CoS-to-DSCP map to generate the internal DSCP value. Layer 2 ISL frame headers carry the CoS value in the three least-significant bits of the one-byte User field. Layer 2 802.1Q frame headers carry the CoS value in the three most significant bits of the Tag Control Information (TCI) field. CoS values range from 0 for low priority to 7 for high priority.
- The trust-DSCP and trust-IP precedence configurations are meaningless for non-IP traffic. If a port is configured with either of these options and non-IP traffic is received, the switch assigns the default port CoS value and generates the internal DSCP from the CoS-to-DSCP map.
- Perform the classification based on the configured Layer 2 MAC ACL, which can examine the MAC source address, the MAC destination address, and the EtherType field. If no ACL is configured, the packet is assigned the default DSCP of 0, which means best-effort traffic; otherwise, the policy map specifies the DSCP to assign to the incoming frame.

■ IP traffic:

- Trust the IP DSCP in the incoming packet (configure the port to trust DSCP), and assign the same DSCP to the packet for internal use. The Internet Engineering Task Force (IETF) defines the six most significant bits of the one-byte ToS field as the DSCP. The priority represented by a particular DSCP value is configurable. DSCP values range from 0 to 63.
- Trust the IP precedence in the incoming packet (configure the port to trust IP precedence), and generate a DSCP by using the configurable IP precedence-to-DSCP map. The IPv4 specification defines the three most significant bits of the one-byte ToS field as the IP precedence. IP precedence values range from 0 for low priority to 7 for high priority.
- Trust the CoS value (if present) in the incoming packet, and generate the DSCP by using the CoS-to-DSCP map.
- Perform the classification based on a configured IP standard or an extended ACL, which examines various fields in the IP header. If no ACL is configured, the packet is assigned the default DSCP of 0, which means best-effort traffic; otherwise, the policy map specifies the DSCP to assign to the incoming frame.

■ Class maps and policy maps:

- A class map is a mechanism used to name and isolate a specific traffic flow, or class, from all other traffic. The class map defines the criteria that is used to match against a specific traffic flow to further classify it; the criteria can include matching the access group defined by the ACL, matching a specific list of DSCP or IP precedence values, or matching a specific list of VLAN IDs associated with another class map that defines the actual criteria (for example, to match a standard or extended ACL). If there is more than one type of traffic that needs to be classified, another class map is created using a different name. After a packet is matched against the class-map criteria, it is further classified through the use of a policy map.
- A policy map specifies which traffic class to act on. Actions can include trusting the CoS, DSCP, or IP precedence values in the traffic class; setting a specific DSCP or IP precedence value in the traffic class; or specifying the traffic bandwidth limitations and the action to take when the traffic is out of profile. Before a policy map can be effective, it must be attached to an interface.

Catalyst 2970 and 3560

- The default CoS-to-DSCP, IP precedence-to-DSCP, and DSCP-to-CoS maps for all Catalyst 2970 and 3560 Switches are identical to those for the Catalyst 3550 Switch, as are mapping functions and syntax, including DSCP mutation.

Classification and Marking on Catalyst Switches (Cont.)

	2950	2960
Trust Capabilities	CoS DSCP --- IP phone Extend trust to IP phone	CoS DSCP IP precedence IP phone Extend trust to IP phone
CoS to DSCP and DSCP to CoS Mapping Tables	Yes	Yes
IPP to DSCP Mapping Table	No	Yes
DSCP Options (pass-thru, mutation)	Yes (no mutation)	Yes
ACL	Yes (no mutation)	Yes (no port range)
Class-Based Markings	Yes	Yes

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.7

Catalyst 2950

Cisco Catalyst 2950 Series Switches offer superior and highly granular QoS based on Layer 2 through Layer 4 information to ensure that network traffic is classified and prioritized, and that congestion is avoided in the best possible manner.

Cisco Catalyst 2950 series switches can classify, reclassify, police (determine if the packet is in or out of predetermined profiles and affect actions on the packet), and mark or drop incoming packets before the packets are placed in the shared buffer. Packet classification allows the network elements to discriminate between various traffic flows and enforce policies based on Layer 2 and Layer 3 QoS fields.

The QoS implementation is based on the DiffServ architecture, an IETF standard. The DiffServ architecture specifies that each packet is classified upon entry into the network. The classification is carried in the IP packet header, using six bits from the deprecated IP ToS field to carry the classification information.

Classification can be carried out using prioritization values in the Layer 2 frame information, as follows:

- Layer 2 802.1Q frame headers are used in trunks, except for native VLAN frames.
- Other non-802.1Q frame types cannot carry Layer 2 CoS values.

Classification can also be carried out using prioritization bits in the Layer 3 IP packets, with DSCP values 0, 8, 10, 16, 18, 24, 26, 32, 34, 40, 46, 48, and 56 only.

Note The Catalyst 2950 Switch does not support VLAN-based classification.

The Catalyst 2950 switch supports four egress queues, which allow you to be discriminating in assigning priorities for the various applications on the LAN. Strict-priority scheduling helps ensure that time-sensitive applications such as voice always follow an expedited path through the switch fabric. WRR scheduling, another significant enhancement, ensures that lower-priority traffic receives attention without comprising the priority settings administered by a network manager. These features allow you to prioritize mission-critical, time-sensitive traffic such as voice (IP telephony traffic); enterprise resource planning (ERP) applications, such as Oracle, SAP, and so on; and computer-aided design (CAD) and computer-aided manufacturing (CAM) over less time-sensitive applications such as FTP or email (Simple Mail Transfer Protocol [SMTP]).

Actions at the egress interface include queuing and scheduling. Queuing evaluates the CoS value and determines which of the four egress queues to place the packet in. Scheduling services the four egress queues based on their configured WRR.

The Catalyst 2950 supports packet classification based on QoS ACLs, as follows:

- IP standard, IP extended, and Layer 2 MAC ACLs can be used to define a group of packets with the same characteristics (class). In the QoS context, the permit and deny actions in the access control entries (ACEs) have different meanings than with security ACLs.
- If a match with a permit action is encountered (first-match principle), the specified QoS-related action is taken.
- If no match with a permit action is encountered and all the ACEs have been examined, no QoS processing occurs on the packet.
- If multiple ACLs are configured on an interface, the packet matches the first ACL with a permit action, and QoS processing begins.
- Configuration of a deny action is not supported in QoS ACLs on the switch.

After a traffic class has been defined with the ACL, a policy can be attached to it. A policy might contain multiple classes with actions specified for each one of them. A policy might include commands to classify the class as a particular aggregate (for example, assign a DSCP) or to rate-limit the class. The policy is then attached to a particular port on which it becomes effective.

You can implement IP ACLs to classify IP traffic by using the **access-list** global configuration command. You can implement Layer 2 MAC ACLs to classify Layer 2 traffic by using the **mac access-list extended** global configuration command.

In the case of frames that arrive without a CoS value (untagged frames), these switches support classification based on a default CoS value per port assigned by the network administrator. After the frames have been classified or reclassified using one of the above modes, the frames are assigned to the appropriate queue at the egress port.

Note	To use the features described in this chapter, you must have the Enhanced Image (EI) installed on your switch.

Catalyst 2960

The Cisco Catalyst 2960 Series Switches provide the following classification options for non-IP traffic:

- Trust the CoS value in the incoming frame by configuring the port to trust CoS. Then use the configurable CoS-to-DSCP map to generate a DSCP value for the packet. If you configure a port to trust DSCP or IP precedence values and non-IP traffic is received, the switch assigns a CoS value and generates an internal DSCP value from the CoS-to-DSCP map. The switch uses the internal DSCP value to generate a CoS value representing the priority of the traffic.
- Perform the classification based on a configured Layer 2 MAC access control list (ACL), which can examine the MAC source address, the MAC destination address, and other fields. If no ACL is configured, the packet is assigned 0 as the DSCP and CoS values, which means best-effort traffic. Otherwise, the policy map action specifies a DSCP or CoS value to assign to the incoming frame.

For IP traffic, Cisco Catalyst 2960 Series Switches provide the following classification options:

- Trust the DSCP value in the incoming packet by configuring the port to trust DSCP. For ports that are on the boundary between two QoS administrative domains, you can modify the DSCP to another value by using the configurable DSCP-to-DSCP mutation map.
- Trust the IP precedence value in the incoming packet by configuring the port to trust IP precedence, and generate a DSCP value for the packet by using the configurable IP-precedence-to-DSCP map.
- Trust the CoS value (if present) in the incoming packet, and generate a DSCP value for the packet by using the CoS-to-DSCP map. If the CoS value is not present, use the default port CoS value.
- Perform the classification based on a configured IP standard or an extended ACL, which examines various fields in the IP header. If no ACL is configured, the packet is assigned 0 as the DSCP and CoS values, which means best-effort traffic. Otherwise, the policy-map action specifies a DSCP or CoS value to assign to the incoming frame.

Cisco Catalyst 2960 Series Switches support traffic policing policies on physical ports. In policy maps on physical ports, you can create these types of policers:

- **Individual:** QoS applies the bandwidth limits specified in the policer separately to each matched traffic class.
- **Aggregate:** QoS applies the bandwidth limits specified in an aggregate policer cumulatively to all matched traffic flows.

Note To use policy maps, the switch must be running the LAN Base image.

Each policer decides on a packet-by-packet basis whether the packet is in or out of profile and specifies the actions on the packet. These actions, carried out by the marker, include passing the packet without modification, dropping the packet, or modifying (marking down) the assigned DSCP of the packet and then allowing the packet to pass through. Cisco Catalyst 2960 Series Switches support out-of-profile markdown for packets that exceed bandwidth utilization limits.

To support egress and ingress queuing, Cisco Catalyst 2960 Series Switches must run the LAN Base image. This switch supports two configurable ingress queues, which are serviced by SRR in shared mode only. One of these queues can be the priority queue. Each port on the Cisco Catalyst 2960 Series Switches supports four egress queues, one of which (queue 1) can be the egress expedite queue. The egress queues are serviced by SRR in either shaping or sharing mode. Both the ingress and egress queues use WTD as the congestion-avoidance mechanism.

Example: Configure Trust Settings on the Cisco Catalyst 2950 Series Switches

Here are three examples showing how to configure the trust setting on the Catalyst 2950 Switch and how each trust setting affects the internal DSCP value, the egress queue, and the egress CoS/DSCP values.

The first example is to trust the incoming CoS, the second example is to trust the incoming DSCP, and the third example is to trust the incoming CoS and pass through the incoming DSCP.

■ Trust CoS:

```
incoming CoS > cos-dscp map > internal dscp > dscp-cos map > egress queue
                                     |
                                     |
                                     | -----> egress dscp
```

■ Trust DSCP:

```
incoming dscp > internal dscp > dscp-cos map > egress queue
                                     |
                                     | -----> egress dscp
```

■ Trust CoS (pass through DSCP):

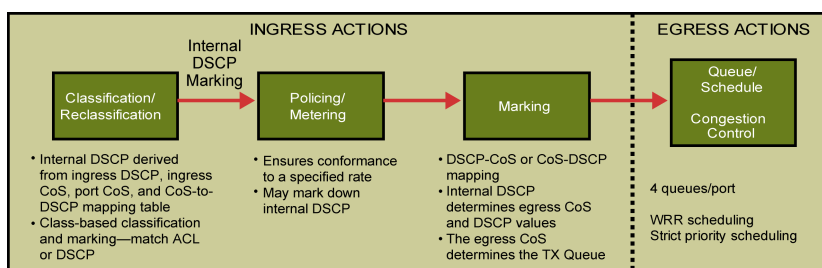
```
incoming CoS > cos-dscp map > internal dscp > dscp-cos map > egress queue
                                     |
                                     | -----> egress dscp
incoming dscp -----> egress dscp
```


Example: QoS Mechanisms on the Cisco Catalyst 2950 Switch

The example in the illustration provides a block diagram of the QoS mechanisms on the Catalyst 2950 Switch.

Catalyst 2950: Aggregate QoS Model

- QoS ACLs using Layer 2, 3, and 4 access control parameters
 - Source/destination MAC address, 16-bit ethertype, source/destination IP address, TCP/UDP source or destination port number
- QoS based on DSCP classification; Support for 13 widely used, well-known DSCP values (0, 8, 10, 16, 18, 24, 26, 32, 34, 40, 46, 48, and 56)
- CoS override per port

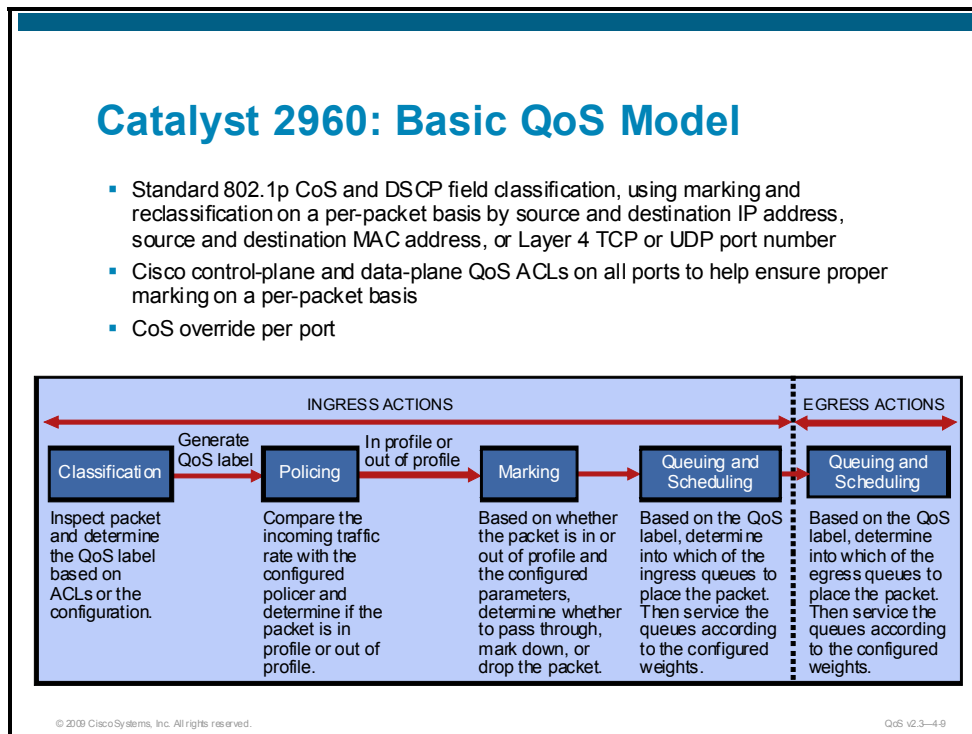


On incoming packets, classification and reclassification are performed by identifying packet groups using either DSCP or CoS. Policing and metering, if configured, are then performed on the packets to ensure compliance to configured rates. Marking is the last action performed on incoming packets, based on the CoS-to-DSCP or DSCP-to-CoS mappings.

Outgoing packets are scheduled and queued for congestion control. There are four queues per port and packets are scheduled based on WRR and strict priority scheduling.

Example: QoS Mechanisms on Cisco Catalyst 2960 Series Switches

The example in the illustration provides a block diagram of the QoS mechanisms on Cisco Catalyst 2960 Series switches.



The figure shows the basic QoS model for Cisco Catalyst 2960 Series Switches. Actions at the ingress port include classifying traffic, policing, marking, queuing, and scheduling, as follows:

- Classifying a distinct path for a packet by associating it with a QoS label. The switch maps the CoS or DSCP in the packet to a QoS label to distinguish one kind of traffic from another. The QoS label that is generated identifies all future QoS actions to be performed on this packet.
- Policing determines whether a packet is in or out of profile by comparing the rate of the incoming traffic to the configured policer. The policer limits the bandwidth consumed by a flow of traffic. The result is passed to the marker.
- Marking evaluates the policer and configuration information for the action to be taken when a packet is out of profile and determines what to do with the packet (pass through a packet without modification, mark down the QoS label in the packet, or drop the packet).
- Queuing evaluates the QoS label and the corresponding DSCP or CoS value to select into which of the two ingress queues to place a packet. Queuing is enhanced with the WTD algorithm, a congestion-avoidance mechanism. If the threshold is exceeded, the packet is dropped.
- Scheduling services the queues based on their configured SRR weights. One of the ingress queues is the priority queue, and SRR services it for its configured share before servicing the other queue.

Actions at the egress port include queuing and scheduling:

- Queuing evaluates the QoS packet label and the corresponding DSCP or CoS value before selecting which of the four egress queues to use. Because congestion can occur when multiple ingress ports simultaneously send data to an egress port, WTD differentiates traffic classes and subjects the packets to different thresholds based on the QoS label. If the threshold is exceeded, the packet is dropped.
- Scheduling services the four egress queues based on their configured SRR shared or shaped weights. One of the queues (queue 1) can be the expedited queue, which is serviced until empty before the other queues are serviced.

Default QoS Configuration: Catalyst 2950 and 3550 Switches

- The default port CoS value is 0.
- The default port trust state is “untrusted.”
- The CoS value of 0 is assigned to all incoming packets.
- Default CoS assignment to priority queues is:
 - CoS 6 to 7: Queue 4
 - CoS 4 to 5: Queue 3
 - CoS 2 to 3: Queue 2
 - CoS 0 to 1: Queue 1
- Default CoS assignment can be altered during configuration.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-10

The default QoS settings for the Catalyst 2950 and 3550 Switches are as follows:

- The default port CoS value is 0.
- The CoS value of 0 is assigned to all incoming packets.
- The default port trust state is untrusted. If a port is connected to an IP phone, you should change the default port configuration to trust the CoS setting from the IP phone using the **mls qos trust** command.
- No policy maps are configured.
- No policers are configured.
- The default CoS assignments to priority queues are as follows:
 - CoS 6 to 7: Queue 4
 - CoS 4 to 5: Queue 3
 - CoS 2 to 3: Queue 2
 - CoS 0 to 1: Queue 1

The default CoS assignments to priority queues change once **auto qos voip** is configured.

- With **auto qos voip [cisco-phone]** or **[trust]**, the default CoS assignments change to:
 - CoS 6 and 7: Queue 3
 - CoS 5: Queue 4
 - CoS 3: Queue 3
 - CoS 0,1,2, and 4: Queue 1

Default QoS Configuration: Catalyst 2960 Switches

- The default port CoS value is 0.
- The default port trust state is “untrusted.”
- CoS and DSCP values are set to 0 for all incoming packets.
- Ingress and egress queues have separate default settings.
- Default CoS assignment can be altered during configuration.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-11

By default, QoS is disabled on Cisco Catalyst 2960 Series Switches. There is no concept of trusted or untrusted ports because the packets are not modified; the CoS, DSCP, and IP precedence values in the packet are not changed. Traffic is switched in pass-through mode. Packets are switched without any rewrites and classified as best effort without any policing.

When QoS is enabled with the **mls qos** global configuration command and all other QoS settings are at their defaults, traffic is classified as best effort (DSCP and CoS values set to 0) without any policing. No policy maps are configured. The default port trust state on all ports is untrusted.

Mapping Tables: Catalyst 2950 and 3550 Switches

- During QoS processing, the switch represents the priority of all traffic (including non-IP traffic) with an internal DSCP value.
- During classification, QoS uses configurable mapping tables to derive the internal DSCP (a six-bit value) from received CoS value.

Default CoS -to-DSCP Map								
CoS value	0	1	2	3	4	5	6	7
DSCP value	0	8	16	24	32	40	48	56

(2950 and 3550)

- Before the traffic reaches the scheduling stage, QoS uses the configurable DSCP-to-CoS map to derive a CoS value from the internal DSCP value.

Default DSCP-to-CoS Map								
DSCP values	0	8,10	16,18	24,26	32,34	40,46	48	56
CoS values	0	1	2	3	4	5	6	7

(2950)

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-12

Actions at the egress interface include queuing and scheduling:

- **Queuing:** Queuing evaluates the internal DSCP and determines which of the four egress queues should be used for placing the packet. The DSCP value is mapped to a CoS value, which selects one of the queues.
- **Scheduling:** Scheduling services the four egress queues based on their configured WRR weights and thresholds. One of the queues can be the expedite queue, which is serviced until empty before the other queues are serviced. Congestion avoidance techniques include tail drop and WRED on Gigabit-capable Ethernet ports and tail drop (with only one threshold) on 10/100 Ethernet ports.

During QoS processing, the switch represents the relative-priority of all traffic (including non-IP traffic) with an internal DSCP value. During classification, QoS uses configurable mapping tables to derive the internal DSCP (a six-bit value) from received CoS or IP precedence (three-bit) values. These maps include the CoS-to-DSCP map and the IP-precedence-to-DSCP map.

On an ingress interface configured in the DSCP-trusted state, if the DSCP values are different between the QoS domains, the configurable DSCP-to-DSCP-mutation map can be applied to the interface that is on the boundary between the two QoS domains.

During policing, QoS can assign another DSCP value to an IP or non-IP packet, if the packet is out of profile and the policer specifies a marked-down DSCP value. This configurable map is called the policed-DSCP map.

Before the traffic reaches the scheduling stage, QoS uses the configurable DSCP-to-CoS map to derive a CoS value from the internal DSCP value. Through the CoS-to-egress-queue map, the CoS values select one of the four egress queues for output processing.

The CoS-to-DSCP, DSCP-to-CoS, and the IP-precedence-to-DSCP (Catalyst 3550 only) maps have default values that might or might not be appropriate for the network.

The default DSCP-to-DSCP-mutation (Catalyst 3550 only) map and the default policed-DSCP map are null maps; they map an incoming DSCP value to the same DSCP value.

The DSCP-to-DSCP-mutation map is the only map that can be applied to a specific Gigabit-capable Ethernet port or to a group of 10/100 Ethernet ports. All other maps apply to the entire switch.

Mapping Tables: Catalyst 2960 Switches

- During QoS processing, the switch represents the priority of all traffic (including non-IP traffic) with a QoS label based on the DSCP or CoS value from the classification stage.
- During classification, QoS uses configurable mapping tables to derive a corresponding DSCP or CoS value from a received CoS, DSCP, or IP precedence value; these maps include the CoS-to-DSCP map and the IP-precedence-to-DSCP map.

Default CoS-to-DSCP Map

CoS value	0	1	2	3	4	5	6	7
DSCP value	0	8	16	24	32	40	48	56

Default IP-Precedence-to-DSCP Map

IP precedence value	0	1	2	3	4	5	6	7
DSCP value	0	8	16	24	32	40	48	56

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-13

During QoS processing, Cisco Catalyst 2960 Series Switches represent the priority of all traffic (including non-IP traffic) with a QoS label based on the DSCP or CoS value from the classification stage.

During classification, QoS uses configurable mapping tables to derive a corresponding DSCP or CoS value from a received CoS, DSCP, or IP precedence value. These maps include the CoS-to-DSCP map and the IP-precedence-to-DSCP map. You can configure these maps by using the **mls qos map cos-dscp** and the **mls qos map ip-prec-dscp** global configuration commands.

On an ingress port configured in the DSCP-trusted state, if the DSCP values are different between the QoS domains, you can apply the configurable DSCP-to-DSCP-mutation map to the port that is on the boundary between the two QoS domains. You can configure this map by using the **mls qos map dscp-mutation** global configuration command.

During policing, QoS can assign another DSCP value to an IP or a non-IP packet (if the packet is out of profile and the policer specifies a marked-down value). This configurable map is called the policed-DSCP map. You can configure this map by using the **mls qos map policed-dscp** global configuration command.

Mapping Tables: Catalyst 2960 Switches (Cont.)

- Before the traffic reaches the scheduling stage, QoS stores the packet in an ingress and an egress queue according to the QoS label; the QoS label is based on the DSCP or the CoS value in the packet and selects the queue through the **DSCP input and output queue threshold maps** or through the **CoS input and output queue threshold maps**.

Default DSCP Input Queue Threshold Map			
DSCP value	0-39	40-47	48-63
Queue ID-Threshold ID	1-1	2-1	1-1

Default DSCP Output Queue Threshold Map					
DSCP value	0-15	16-31	32-39	40-47	48-63
Queue ID-Threshold ID	2-1	3-1	4-1	1-1	4-1

Default CoS Input Queue Threshold Map			
CoS value	0-4	5	6, 7
Queue ID-Threshold ID	1-1	2-1	1-1

Default CoS Output Queue Threshold Map					
CoS value	0, 1	2, 3	4	5	6, 7
Queue ID-Threshold ID	2-1	3-1	4-1	1-1	4-1

© 2009 Cisco Systems, Inc. All rights reserved.

QoS V2.3-4-14

Before the traffic reaches the scheduling stage, QoS stores the packet in an ingress and an egress queue according to the QoS label. The QoS label is based on the DSCP or the CoS value in the packet and selects the queue through the DSCP input and output queue threshold maps or through the CoS input and output queue threshold maps. In addition to an ingress or an egress queue, the QoS label also identifies the WTD threshold value. You can configure these maps by using the **mls qos srr-queue {input | output} dscp-map** and the **mls qos srr-queue {input | output} cos-map** global configuration commands.

The DSCP-to-CoS map is used to generate a CoS value, which is used to select one of the four egress queues.

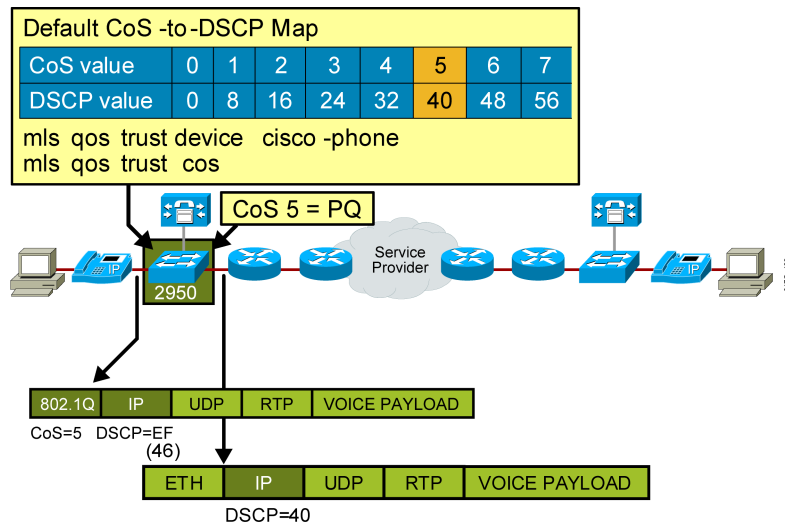
Default DSCP-to-CoS Map

DSCP Value	CoS Value
0-7	0
8-15	1
16-23	2
24-31	3
32-39	4
40-47	5
48-55	6
56-63	7

The CoS-to-DSCP, DSCP-to-CoS, and the IP-precedence-to-DSCP maps have default values that might or might not be appropriate for your network.

The default DSCP-to-DSCP-mutation map and the default policed-DSCP map are null maps; they map an incoming DSCP value to the same DSCP value. The DSCP-to-DSCP-mutation map is the only map you apply to a specific port. All other maps apply to the entire switch.

Mapping Tables Example 1: Life of a High-Priority (VoIP) Packet

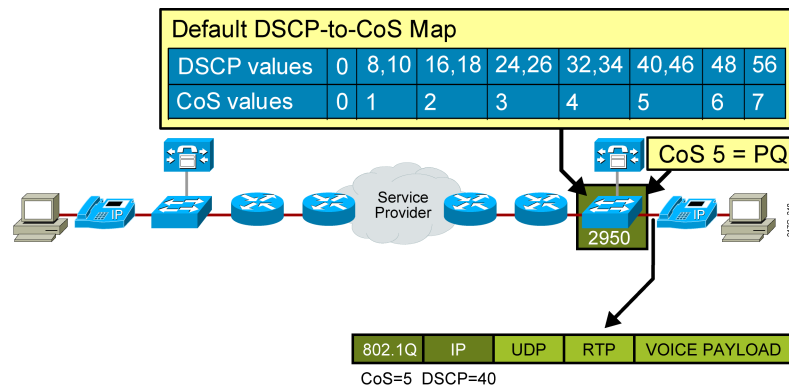


The figure provides an example of a CoS value mapped to the DSCP value in a Catalyst 2950 Switch.

The trust boundary has been established on the switch port to trust the CoS setting from the IP phone. By default, the CoS and DSCP value of a packet coming from a Cisco IP phone is set to CoS 5 and DSCP Expedited Forwarding (EF) value 46.

On the output of the switch, in the Layer 3 header, the DSCP will be set to 40 using the default CoS-to-DSCP map.

Mapping Tables Example 2: Life of a High-Priority (VoIP) Packet



© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-16

The figure shows the previous packet as it arrives to its destination after traversing the network.

In this example, the switch port connecting to the router is set to trust DSCP. Therefore, the Layer 3 header will have a DSCP value of 40 (from the previous slide) and as it traverses the switch, its CoS value is set to 5 using the default DSCP-to-CoS map.

Configuring LAN-Based Classification and Marking

This topic describes the Cisco IOS commands that are required to configure LAN-based classification and marking.

Configuring Classification and Marking on Catalyst 2950 Switches

Switch(config-if)#

```
mls qos trust [cos [pass-through dscp] | device cisco-  
phone | dscp]
```

- Configures the port to trust state on an interface.
- When a port is configured with trust DSCP and the incoming packet is a tagged non-IP packet, the CoS value for the packet is set to 0, and the DSCP-to-CoS map is not applied.
- If DSCP is trusted, the DSCP field of the IP packet is not modified, but it is still possible that the CoS value of the packet is modified according to the DSCP-to-CoS map.

Switch(config-if)#

```
mls qos cos {default-cos | override}
```

- Defines the default CoS value of a port or assigns the default CoS to all incoming packets on the port.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-17

The figure shows some of the QoS configuration commands that are necessary for Catalyst 2950 switches. The defaults for its interfaces are as follows:

- The port is not trusted.
- Pass-through mode is disabled.
- Trusted boundary is disabled.

The syntax for the **mls qos trust** command is as follows:

```
mls qos trust [cos [pass-through dscp] | device cisco-phone |  
dscp]
```

Syntax Description

Parameter	Description
cos	(Optional) Specifies that the CoS bits in incoming frames are trusted and derives the internal DSCP value from the CoS bits.
pass-through dscp	(Optional) Configures the interface to classify ingress packets by trusting the CoS value and to send packets without modifying the DSCP value (pass-through mode).
device cisco-phone	(Optional) Classifies ingress packets by trusting the value sent from the Cisco IP phone (trusted boundary).
dscp	(Optional) Classifies ingress packets with packet DSCP values (most significant 6 bits of the 8-bit service-type field). For non-IP packets, the packet CoS value is set to 0. This keyword is available only if your switch is running the EI software.

To define the default CoS value for an interface, use the **mls qos cos** interface configuration command. Use the **no** form of this command to remove a prior entry. QoS assigns the CoS value specified with this command to untagged frames received on trusted and untrusted ports. The default CoS value is 0.

The syntax for the **mls qos cos** command is as follows:

```
mls qos cos cos-value
```

Syntax Description

Parameter	Description
<i>cos-value</i>	Default CoS value for the interface; valid values are from 0 to 7.

The syntax for the **mls qos trust** command in Cisco IOS Software Release 12.2(46)SE on Cisco Catalyst 2960 Series Switches is as follows:

```
mls qos trust [cos | device cisco-phone | dscp | ip-  
precedence]
```

Syntax Description

Parameter	Description
cos	(Optional) Classifies an ingress packet by using the packet CoS value. For an untagged packet, use the port default CoS value.
device cisco-phone	(Optional) Classifies an ingress packet by trusting the CoS or DSCP value sent from the Cisco IP phone (trusted boundary), depending on the trust setting.
dscp	(Optional) Classifies an ingress packet by using the packet DSCP value (most significant 6 bits of 8-bit service-type field). For a non-IP packet, the packet CoS is used if the packet is tagged. For an untagged packet, the default port CoS value is used.
ip-precedence	(Optional) Classifies an ingress packet by using the packet IP-precedence value (most significant 3 bits of 8-bit service-type field). For a non-IP packet, the packet CoS is used if the packet is tagged. For an untagged packet, the port default CoS value is used.

The syntax shown in the figure for the **mls qos cos** command is the same as the syntax for the **mls qos cos** command in Cisco IOS Software Release 12.2(46)SE on Cisco Catalyst 2960 Series Switches.

Configuring Classification and Marking on Catalyst 2950 Switches (Cont.)

Switch(config)#

```
mls qos map cos-dscp dscp1...dscp8
```

- Defines the CoS-to-DSCP mapping.
- For *dscp1...dscp8*, enter eight DSCP values that correspond to CoS values 0 to 7. Separate each DSCP value with a space.
- The supported DSCP values are 0, 8, 10, 16, 18, 24, 26, 32, 34, 40, 46, 48, and 56.

Switch(config)#

```
mls qos map dscp-cos dscp-list to cos
```

- Defines the DSCP-to-CoS mapping.
- For *dscp-list*, enter up to 13 DSCP values separated by spaces. Then enter the **to** keyword. The supported DSCP values are 0, 8, 10, 16, 18, 24, 26, 32, 34, 40, 46, 48, and 56.
- For **cos**, enter the CoS value to which the DSCP values correspond. The CoS range is 0 to 7.

© 2009 Cisco Systems, Inc. All rights reserved.

CoS v2.3-4-18

The commands listed in the figure show how to change the default CoS-to-DSCP and DSCP-to-CoS mappings.

CoS-to-DSCP Default Mapping

Marker	Value							
CoS Values	0	1	2	3	4	5	6	7
DSCP Values	0	8	16	24	32	40	48	56

To define the ingress CoS-to-DSCP mapping for trusted interfaces, use the **mls qos map cos-dscp** command. The CoS-to-DSCP map is used to map the CoS of packets arriving on trusted interfaces (or flows) to a DSCP where the trust type is trust-cos. This map is a table of eight CoS values (0 through 7) and their corresponding DSCP values. Use the **no** form of this command to remove a prior entry.

The syntax for the **mls qos map cos-dscp** command is as follows:

mls qos map cos-dscp *values*

Syntax Description

Parameter	Description
<i>values</i>	Eight DSCP values, separated by spaces, corresponding to the CoS values; valid values are from 0 to 63.

DSCP-to-CoS Default Mapping

Marker	Value							
DSCP Values	0	8, 10	16, 18	24, 26	32, 34	40, 46	48	56
CoS Values	0	1	2	3	4	5	6	7

To define an egress DSCP-to-CoS mapping, use the **mls qos map dscp-cos** command. The DSCP-to-CoS map is used to map DSCP values in incoming packets to a CoS value, which is used to select one of the four egress queues. The CoS mapped value is written into the ISL header or 802.1Q tag of the transmitted frame on trunk interfaces. You can enter up to eight DSCP values separated by a space, and up to eight CoS values separated by a space. Use the **no** form of this command to remove a prior entry.

The syntax for the **mls qos map dscp-cos** command is as follows:

mls qos map dscp-cos *dscp-values to cos-values*

Syntax Description

Parameter	Description
<i>dscp-values</i>	DSCP values; valid values are from 0 to 63.
to	Defines mapping.
<i>cos-values</i>	CoS values; valid values are from 0 to 63.

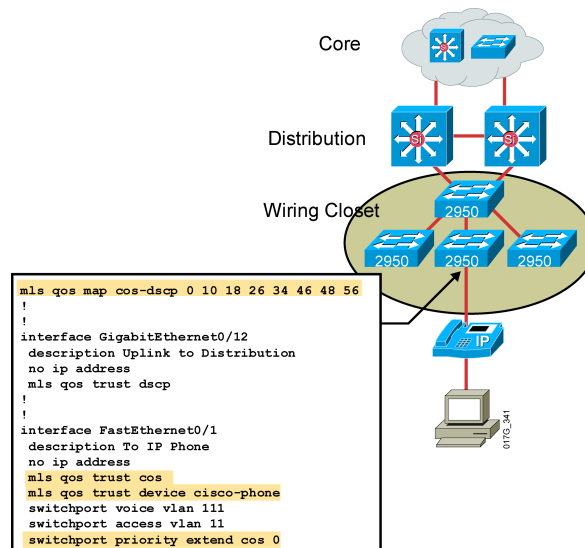
The syntax for the **mls qos map** command in Cisco IOS Software Release 12.2(46)SE on Cisco Catalyst 2960 Series Switches is as follows:

```
mls qos map {cos-dscp dscp1...dscp8 | dscp-cos dscp-list to  
cos | dscp-mutation dscp-mutation-name in-dscp to out-dscp |  
ip-prec-dscp dscp1...dscp8 | policed-dscp dscp-list to mark-  
down-dscp}
```

Syntax Description

Parameter	Description
cos-dscp <i>dscp1...dscp8</i>	Defines the CoS-to-DSCP map. For <i>dscp1...dscp8</i> , enter eight DSCP values that correspond to CoS values 0 to 7. Separate each DSCP value with a space. The range is 0 to 63.
dscp-cos <i>dscp-list</i> to <i>cos</i>	Defines the DSCP-to-CoS map. For <i>dscp-list</i> , enter up to eight DSCP values, with each value separated by a space. The range is 0 to 63. Then enter the to keyword. For <i>cos</i> , enter a single CoS value to which the DSCP values correspond. The range is 0 to 7.
dscp-mutation <i>dscp-mutation-name</i> <i>in-dscp</i> to <i>out-dscp</i>	Defines the DSCP-to-DSCP-mutation map. For <i>dscp-mutation-name</i> , enter the mutation map name. For <i>in-dscp</i> , enter up to eight DSCP values, with each value separated by a space. Then enter the to keyword. For <i>out-dscp</i> , enter a single DSCP value. The range is 0 to 63.
ip-prec-dscp <i>dscp1...dscp8</i>	Defines the IP-precedence-to-DSCP map. For <i>dscp1...dscp8</i> , enter eight DSCP values that correspond to the IP precedence values 0 to 7. Separate each DSCP value with a space. The range is 0 to 63.
policed-dscp <i>dscp-list</i> to <i>mark-down-dscp</i>	Defines the policed-DSCP map. For <i>dscp-list</i> , enter up to eight DSCP values, with each value separated by a space. Then enter the to keyword. For <i>mark-down-dscp</i> , enter the corresponding policed (marked down) DSCP value. The range is 0 to 63.

Configuring Classification and Marking on Catalyst 2950 Switches (Cont.)



The figure shows a configuration example on a Catalyst 2950 Switch where the CoS-to-DSCP map has been changed from the default.

The default map is:

Marker	Value							
CoS Values	0	1	2	3	4	5	6	7
DSCP Values	0	8	16	24	32	40	48	56

And the map after configuration is:

Marker	Value							
CoS Values	0	1	2	3	4	5	6	7
DSCP Values	0	10	18	26	34	46	48	56

The interface has been set to trust the CoS value using the **mls qos trust** command, using both the **cos** and **cisco-phone** options. The result of the configuration is that the switch interface trusts CoS only when a Cisco IP phone is attached. The switch uses Cisco Discovery Protocol to detect whether a Cisco IP phone is attached, and also passes the voice VLAN ID information to the Cisco IP phone using Cisco Discovery Protocol.

Use the **switchport priority extend cos 0** interface configuration command to enable the IP phone to override the CoS marking from the PC attached to the IP phone with a CoS value of 0.

Use the **switchport priority extend** interface configuration command to set a port priority for the incoming untagged frames or the priority of frames received by the IP phone connected to the specified port. Use the **no** form of this command to return to the default setting.

The syntax for the **switchport priority extend** command is as follows:

switchport priority extend {*cos value* | **trust**}

Syntax Description

Parameter	Description
<i>cos value</i>	Sets the IP phone port to override the priority received from PC or the attached device. The CoS value is a number from 0 to 7. 7 is the highest priority. The default is 0.
trust	Sets the IP phone port to trust the priority received from PC or the attached device.

The syntax for the **switchport priority extend** command is the same as the syntax for the **switchport priority extend** command in Cisco IOS Software Release 12.2(46)SE on Cisco Catalyst 2960 Series Switches.

Configuring Classification and Marking on Catalyst 2950 Switches (Cont.)

Classification and marking can also be performed using MQC (class maps and policy maps):

1. Create an IP standard or extended ACL for IP traffic, or a Layer 2 MAC ACL for non-IP traffic.
2. Create a class map and define the match criteria to classify traffic.
3. Create a service policy to perform the appropriate QoS action (mark, police, and so on).
4. Apply the service policy to a switch interface.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS V2.3-4-20

Create a class map by using the **class-map** global configuration command. The **class-map** global configuration command is used to isolate a specific traffic flow (or class) from all other traffic and to name it. Use this command when the map is shared among many ports. When you enter the **class-map** global configuration command, the switch enters the class-map configuration mode. In this mode, you define the match criterion for the traffic by using the **match class-map** configuration command. Match statements can only include ACLs.

Create and name a policy map by using the **policy-map** global configuration command. When you enter this command, the switch enters the policy-map configuration mode. In this mode, the actions to take on a specific traffic class are specified by using the **class policy-map** configuration or **set policy-map class** configuration command. To make the policy map effective, you attach it to an interface by using the **service-policy interface** configuration command.

A policy map also has these characteristics:

- A policy map can contain multiple class statements, each with different match criteria and policers.
- A separate policy-map class can exist for each type of traffic received through an interface.
- Only one policy map per interface can be attached in the input direction.

Note This procedure is the same in Cisco IOS Software Release 12.2(46)SE on Cisco Catalyst 2960 Series Switches.

Configuring Classification and Marking on Catalyst 2950 Switches (Cont.)

Switch(config) #

```
access-list access-list-number {deny | permit | remark remark}  
{source source-wildcard | host source | any}
```

- Configures a standard IP access control list that is based on source address only.
- The default standard ACL is always terminated by an implicit deny statement for all packets.

Switch(config) #

```
access-list access-list-number {deny | permit | remark remark}  
protocol {source source-wildcard | host source | any} [operator  
port] {destination destination-wildcard | host destination | any}  
[operator port] [dscp dscp-value] [time-range time-range-name]
```

- Configures an extended IP access control list that can be based on source, destination, port, DSCP value, or a time range.
- The default extended ACL is always terminated by an implicit deny statement for all packets.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-21

You can use IP standard, IP extended, and Layer 2 MAC ACLs to define a group of packets with the same characteristics (class). In the QoS context, the permit and deny actions in the ACEs have different meanings than with security ACLs, as follows:

- If a match with a permit action is encountered (first-match principle), the specified QoS-related action is taken.
- If no match with a permit action is encountered and all the ACEs have been examined, no QoS processing occurs on the packet.
- If multiple ACLs are configured on an interface, the packet matches the first ACL with a permit action, and QoS processing begins.
- Configuration of a deny action is not supported in QoS ACLs on the switch.

Use the standard version of the **access-list** global configuration command to configure a standard IP ACL. Use the **no** form of this command to remove a standard IP ACL.

The syntax for the standard **access-list** command is as follows:

```
access-list access-list-number {deny | permit | remark remark}  
{source source-wildcard | host source | any}
```

Syntax Description

Parameter	Description
<i>access-list-number</i>	Number of an ACL, from 1 to 99 or from 1300 to 1999.
deny	Denies access if conditions are matched.
permit	Permits access if conditions are matched.
remark <i>remark</i>	ACL entry comment; up to 100 characters.
<i>source</i> <i>source-wildcard</i> host <i>source</i> any	Defines a source IP address and wildcard. The <i>source</i> is the source address of the network or host from which the packet is being sent, specified in one of these ways: ■ The 32-bit quantity in dotted-decimal format. The <i>source-wildcard</i> applies wildcard bits to the source. ■ The keyword host, followed by the 32-bit quantity in dotted-decimal format, as an abbreviation for <i>source</i> and <i>source-wildcard</i> of <i>source</i> 0.0.0.0. ■ The keyword any as an abbreviation for <i>source</i> and <i>source-wildcard</i> of 0.0.0.0 255.255.255.255. You do not need to enter a source-wildcard.

Use the extended version of the **access-list** global configuration command to configure an extended IP ACL. Use the **no** form of this command to remove an extended IP ACL.

```
access-list access-list-number {deny | permit | remark remark}  
protocol {source source-wildcard | host source | any}  
[operator port] {destination destination-wildcard |  
host destination | any} [operator port] [dscp dscp-value]  
[time-range time-range-name]
```

Syntax Description

Parameter	Description
<i>access-list-number</i>	Number of an ACL, from 100 to 199 or from 2000 to 2699.
<i>protocol</i>	Name of an IP protocol. <i>protocol</i> can be ip , tcp , or udp .
deny	Denies access if conditions are matched.
permit	Permits access if conditions are matched.
remark <i>remark</i>	ACL entry comment; up to 100 characters.
<i>source</i> <i>source-wildcard</i> host <i>source</i> any	Defines a source IP address and wildcard. The <i>source</i> is the source address of the network or host from which the packet is being sent, specified in one of these ways: ■ The 32-bit quantity in dotted-decimal format. The <i>source-wildcard</i> applies wildcard bits to the source. ■ The keyword host, followed by the 32-bit quantity in dotted-decimal format, as an abbreviation for <i>source</i> and <i>source-wildcard</i> of <i>source</i> 0.0.0.0. ■ The keyword any as an abbreviation for <i>source</i> and <i>source-wildcard</i> of 0.0.0.0 255.255.255.255. You do not need to enter a source-wildcard.
<i>destination</i> <i>destination-wildcard</i> host <i>destination</i> any	Defines a destination IP address and wildcard. The <i>destination</i> is the destination address of the network or host to which the packet is being sent, specified in one of these ways: ■ The 32-bit quantity in dotted-decimal format. The <i>destination-wildcard</i> applies wildcard bits to the destination. ■ The keyword host, followed by the 32-bit quantity in dotted-decimal format, as an abbreviation for <i>destination</i> and <i>destination-wildcard</i> of <i>source</i> 0.0.0.0. ■ The keyword any as an abbreviation for <i>destination</i> and <i>destination-wildcard</i> of 0.0.0.0 255.255.255.255. You do not need to enter a destination-wildcard.
operator <i>port</i>	(Optional) Defines a source or destination port. The <i>operator</i> can be only eq (equal). If <i>operator</i> is after the source IP address and wildcard, conditions match when the source port matches the defined port. If <i>operator</i> is after the destination IP address and wildcard, conditions match when the destination port matches the defined port. The <i>port</i> is a decimal number or name of a TCP or UDP port. The number can be from 0 to 65535. Use TCP port names only for TCP traffic. Use UDP port names only for UDP traffic.

Parameter	Description
dscp <i>dscp-value</i>	(Optional) Defines a DSCP value to classify traffic. For the <i>dscp-value</i> , enter any of the 13 supported DSCP values (0, 8, 10, 16, 18, 24, 26, 32, 34, 40, 46, 48, and 56), or use the question mark (?) to see a list of available values.
time-range <i>time-range-name</i>	(Optional) For the time-range keyword, enter a meaningful name to identify the time range. For a more detailed explanation of this keyword, refer to the software configuration guide.

The syntax shown in the figure is compatible with Cisco IOS Software Release 12.2(46)SE on Cisco Catalyst 2960 Series Switches. For the command syntax for other software versions or platforms, refer to the appropriate Command Reference on <http://www.cisco.com>.

Configuring Classification and Marking on Catalyst 2950 Switches (Cont.)

Switch(config)#

```
class-map class-map-name
```

- Creates a class map to be used for matching packets.
- Only one match criterion per class map is supported. For example, when defining a class map, only one match command can be entered.

Switch(config-cmap)#

```
match {access-group acl-index | access-group name acl-name  
| ip dscp dscp-list}
```

- Defines the match criteria to classify traffic.
- Only IP access groups, MAC access groups, and classification based on DSCP values are supported.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-22

You can also use modular QoS command-line interface (CLI), or MQC, class maps on Catalyst 2950 switches for packet classification purposes. However, the **match** command used in conjunction with the class map has different parameters when executed on a Catalyst switch.

Use the **match** class-map configuration command to define the match criteria to classify traffic. Use the **no** form of this command to remove the match criteria.

The syntax for the **match** command is as follows:

```
match {access-group acl-index | access-group name acl-name | ip dscp dscp-list}
```

Syntax Description

Parameter	Description
access-group <i>acl-index</i>	Number of an IP standard or extended ACL. For an IP standard ACL, the ACL index range is 1 to 99 and 1300 to 1999. For an IP extended ACL, the ACL index range is 100 to 199 and 2000 to 2699.
access-group name <i>acl-name</i>	Name of an IP standard or extended ACL or name of an extended MAC ACL. Note: The ACL name must begin with an alphabetic character to prevent ambiguity with numbered ACLs. A name also cannot contain a space or quotation mark.
ip dscp <i>dscp-list</i>	List of up to eight DSCP values for each match statement to match against incoming packets. Separate each value with a space. The supported DSCP values are 0, 8, 10, 16, 18, 24, 26, 32, 34, 40, 46, 48, and 56.

The syntax for the **class-map** command in Cisco IOS Software Release 12.2(46)SE on Cisco Catalyst 2960 Series Switches is as follows:

```
class-map [match-all | match-any] class-map-name
```

Syntax Description

Parameter	Description
match-all	(Optional) Performs a logical AND of all matching statements under this class map. All criteria in the class map must be matched.
match-any	(Optional) Performs a logical OR of the matching statements under this class map. One or more criteria must be matched.
<i>class-map-name</i>	Name of the class map.

The syntax for the **match** command in Cisco IOS Software Release 12.2(46)SE on Cisco Catalyst 2960 Series Switches is as follows:

```
match {access-group acl-index-or-name | ip dscp dscp-list | ip  
precedence ip-precedence-list}
```

Syntax Description

Parameter	Description
access-group <i>acl-index-or-name</i>	Number or name of an IP standard or extended ACL or MAC ACL. For an IP standard ACL, the ACL index range is 1 to 99 and 1300 to 1999. For an IP extended ACL, the ACL index range is 100 to 199 and 2000 to 2699.
ip dscp <i>dscp-list</i>	List of up to eight IP DSCP values to match against incoming packets. Separate each value with a space. The range is 0 to 63. You also can enter a mnemonic name for a commonly used value.
ip precedence <i>ip-precedence-list</i>	List of up to eight IP precedence values to match against incoming packets. Separate each value with a space. The range is 0 to 7. You also can enter a mnemonic name for a commonly used value.

Configuring Classification and Marking on Catalyst 2950 Switches (Cont.)

Switch(config)#

```
policy-map policy-map-name
```

- Creates or modifies a policy map that can be attached to multiple interfaces

Switch(config-pmap)#

```
class class-map-name [access-group name acl-index-or-name]
```

- Defines a traffic classification for the policy to act on using the class-map name or access group

Switch(config-pmap-c)#

```
set ip dscp new-dscp
```

- Used to mark packets with a new DSCP value. Supported DSCP values are 0, 8, 10, 16, 18, 24, 26, 32, 34, 40, 46, 48, and 56

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-20

Recall that a policy map creates or modifies a policy that can be attached to multiple interfaces. The **class** command defines traffic classification for the policy to act on based on the class map or the access group. Use the **class** policy-map configuration command to define a traffic classification for the policy to act on using the class map name or access group. Use the **no** form of this command to delete an existing class map.

The syntax for the **class** command is as follows:

```
class class-map-name [access-group name acl-index-or-name]
```

Syntax Description

Parameter	Description
access-group name <i>acl-index-or-name</i>	(Optional) Number or name of an IP standard or extended ACL or name of an extended MAC ACL. For an IP standard ACL, the index range is 1 to 99 and 1300 to 1999; for an IP extended ACL, the index range is 100 to 199 and 2000 to 2699.

Use the **set ip dscp** policy-map class configuration command to classify IP traffic by setting a DSCP value.

```
set ip dscp new-dscp
```

Syntax Description

Parameter	Description
<i>new-dscp</i>	New DSCP value assigned to the classified traffic. The supported DSCP values are 0, 8, 10, 16, 18, 24, 26, 32, 34, 40, 46, 48, and 56.

The syntax shown in the figure for the **policy-map** command is the same as the syntax for the **policy-map** command in Cisco IOS Software Release 12.2(46)SE on Cisco Catalyst 2960 Series Switches.

The syntax for the **class** command in Cisco IOS Software Release 12.2(46)SE on Cisco Catalyst 2960 Series Switches is as follows:

```
class class-map-name
```

The syntax for the **set** command in Cisco IOS Software Release 12.2(46)SE on Cisco Catalyst 2960 Series Switches is as follows:

```
set {dscp new-dscp | [ip] precedence new-precedence}
```

Syntax Description

Parameter	Description
dscp <i>new-dscp</i>	New DSCP value assigned to the classified traffic. The range is 0 to 63. You also can enter a mnemonic name for a commonly used value.
[ip] precedence <i>new-precedence</i>	New IP-precedence value assigned to the classified traffic. The range is 0 to 7. You also can enter a mnemonic name for a commonly used value.

Configuring Classification and Marking on Catalyst 2950 Switches (Cont.)

Switch(config-if)#

```
service-policy input policy-map-name
```

- Applies a policy map defined by the **policy-map** command to the input of a particular interface

```
mac access-list extended maclist1
 permit host 0001.0000.0001 host 0002.0000.0001
!
class-map macclass1
 match access-group name maclist1
!
policy-map macpolicy1
 class macclass1
  set ip dscp 26
!
interface gigabitethernet0/1
 switchport mode trunk
 mls qos trust cos
 service-policy input macpolicy1
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-24

The last step in configuring a policy is to apply the policy to the interface.

In the example, an extended access list has been created for a MAC address, maclist1. A class map, macclass1, has been created that will match any MAC address permitted by the access list maclist1.

If there is a match for the class map macclass1, the DSCP field will be set to 26 as defined in the policy map macpolicy1.

This policy map has been implemented on the Gigabit Ethernet port 0/1 for incoming packets.

The syntax shown in the figure for the **service-policy** command is the same as the syntax for the **service-policy** command in Cisco IOS Software Release 12.2(46)SE on Cisco Catalyst 2960 Series Switches.

Monitoring LAN-Based Classification and Marking

This topic describes some of the Cisco IOS commands required to monitor LAN-based classification and marking.

Monitoring QoS on Catalyst 2950 Switches

```
Switch>
```

```
show mls qos interface [interface-id] [policers]
```

- Displays QoS information at the interface level

```
Switch> show mls qos interface fastethernet0/1

FastEthernet0/1
trust state:trust cos
trust mode:trust cos
COS override:dis
default COS:0
pass-through:none
trust device:cisco-phone
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.35

After QoS has been configured on a Catalyst switch, you should verify proper operation of QoS and verify that the policies have been configured. In the example, the trust state has been set for CoS and the default value of CoS is 0.

Use the **show mls qos interface** user EXEC command to display QoS information at the interface level.

The syntax for the **show mls qos interface** command is as follows:

show mls qos interface [*interface-id*] [**policers**]

Syntax Description

Parameter	Description
<i>interface-id</i>	(Optional) Displays QoS information for the specified interface
policers	(Optional) Displays all the policers configured on the interface, their settings, and the number of policers unassigned (available only when the switch is running the EI software)

The syntax for the **show mls qos interface** command in Cisco IOS Software Release 12.2(46)SE on Cisco Catalyst 2960 Series Switches is as follows:

show mls qos interface [*interface-id*] [**buffers** | **queueing** | **statistics**] [| {**begin** | **exclude** | **include**} *expression*]

Syntax Description

Parameter	Description
<i>interface-id</i>	(Optional) Displays QoS information for the specified port. Valid interfaces include physical ports
buffers	(Optional) Displays the buffer allocation among the queues
queueing	(Optional) Displays the queueing strategy (shared or shaped) and the weights corresponding to the queues
statistics	(Optional) Displays statistics for sent and received DSCP values and CoS values, the number of packets enqueued or dropped per egress queue, and the number of in-profile and out-of-profile packets for each policer
begin	(Optional) Displays begins with the line that matches the expression
exclude	(Optional) Displays excludes lines that match the expression
include	(Optional) Displays includes lines that match the specified expression
<i>expression</i>	Expression in the output to use as a reference point

Monitoring QoS on Catalyst 2950 Switches (Cont.)

Switch>

```
show mls qos maps [cos-dscp | dscp-cos]
```

- Displays QoS mapping information

```
Switch> show mls qos maps

Dscp-cos map:
dscp: 0 8 10 16 18 24 26 32 34 40 46 48 56
-----
cos:  0 1  1  2  2  3  3  4  4  5  5  6  7

Cos-dscp map:
cos:  0 1 2  3  4  5  6  7
-----
dscp: 0 8 16 24 32 40 48 56
```

© 2009 Cisco Systems, Inc. All rights reserved.

QoS V2.3-4.36

Another important monitoring command is the **show mls qos maps** command, which displays the CoS-to-DSCP and DSCP-to-CoS mappings. Maps are used to generate an internal DSCP value which represents the priority of the traffic.

The syntax for the **show mls qos maps** command is as follows:

```
show mls qos maps [cos-dscp | dscp-cos]
```

Syntax Description

Parameter	Description
cos-dscp	(Optional) Displays CoS-to-DSCP map
dscp-cos	(Optional) Displays DSCP-to-CoS map

The syntax for the **show mls qos maps** command in Cisco IOS Software Release 12.2(46)SE on Cisco Catalyst 2960 Series Switches is as follows:

```
show mls qos maps [cos-dscp | cos-input-q | cos-output-q |
dscp-cos | dscp-input-q | dscp-mutation dscp-mutation-name |
dscp-output-q | ip-prec-dscp | policed-dscp] [ | {begin |
exclude | include} expression]
```

Syntax Description

Parameter	Description
cos-dscp	(Optional) Displays CoS-to-DSCP map
cos-input-q	(Optional) Displays the CoS input queue threshold map
cos-output-q	(Optional) Displays the CoS output queue threshold map
dscp-cos	(Optional) Displays DSCP-to-CoS map
dscp-input-q	(Optional) Displays the DSCP input queue threshold map
dscp-mutation <i>dscp-mutation-name</i>	(Optional) Displays the specified DSCP-to-DSCP-mutation map
dscp-output-q	(Optional) Displays the DSCP output queue threshold map
ip-prec-dscp	(Optional) Displays the IP-precedence-to-DSCP map
policed-dscp	(Optional) Displays the policed-DSCP map
begin	(Optional) Displays begins with the line that matches the expression
exclude	(Optional) Displays excludes lines that match the expression
include	(Optional) Displays includes lines that match the specified expression
<i>expression</i>	Expression in the output to use as a reference point

Summary

This topic summarizes the key points discussed in this lesson.

Summary

- QoS classification and marking on workgroup switches are based on DiffServ and CoS. There must be mapping between Layer 2 and Layer 3.
- For most Cisco IP phone configurations, the traffic sent from the telephone to the switch may be trusted to ensure that voice traffic is properly prioritized over other types of traffic in the network.
- Several types of classification and marking are available on Cisco Catalyst 6500, 4000, 3750, 3500, 2950, and 2960 Switches.
- QoS assigns the CoS value specified with **mls qos cos** interface configuration command to untagged frames received on trusted and untrusted ports.
- Use the **show mls qos** interface command to display general QoS information.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4-27

Understanding QoS in the Life of a Packet

Overview

This lesson provides information regarding the application of quality of service (QoS) mechanisms throughout a simple network. The lesson follows two packets—a high-priority voice packet and a low-priority file transfer packet—as they traverse a QoS-enabled network.

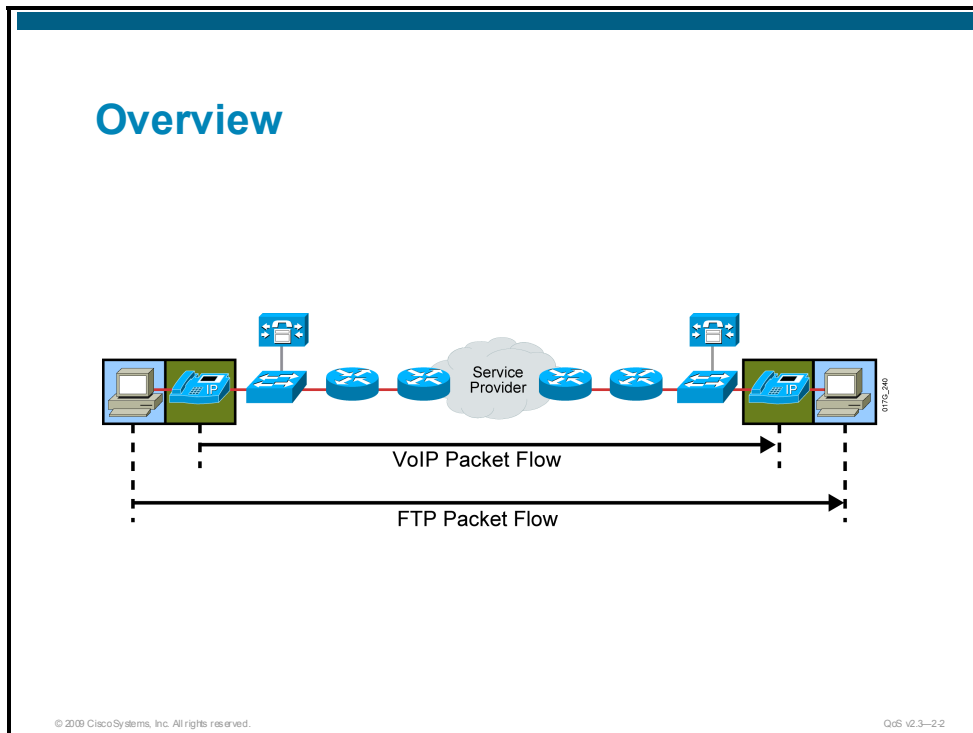
Objectives

Upon completing this lesson, you will be able to describe how QoS mechanisms have an impact on IP packets. This ability includes being able to meet these objectives:

- Describe how QoS mechanisms have an impact on IP packets
- On a network diagram, identify key points where the QoS status of a high-priority VoIP packet can be altered as QoS policies are applied to the IP packet
- On a network diagram, identify key points where the QoS status of a low-priority FTP packet can be altered as QoS policies are applied to the IP packet

QoS and Packets

This topic describes how QoS mechanisms have an impact on IP packets.



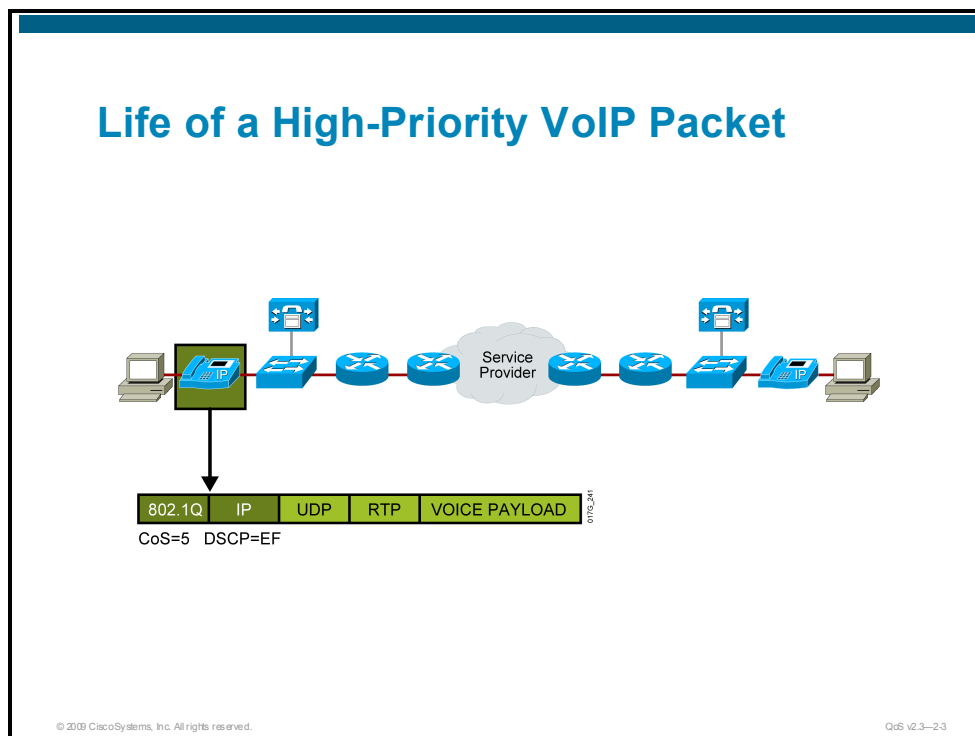
The first packet, packet 1, is a high-priority VoIP packet that will receive preferential treatment as it moves through the network.

The second packet, packet 2, is a low-priority FTP packet that will receive deferential treatment as it moves through the network.

A QoS peering relation between the enterprise and the service provider is assumed. The service provider will recognize and act upon QoS classifications made by the enterprise customer. The relationship shows how QoS can be effectively honored across an enterprise and service provider boundary.

Life of a High-Priority VoIP Packet

This topic describes the life of a high-priority VoIP packet.



As it begins its life in the IP phone, the VoIP packet is immediately marked with both of the following:

- **Layer 2:** 802.1Q CoS = 5 (highest priority in an Ethernet LAN)
- **Layer 3:** DSCP = EF (highest priority in an IP network)

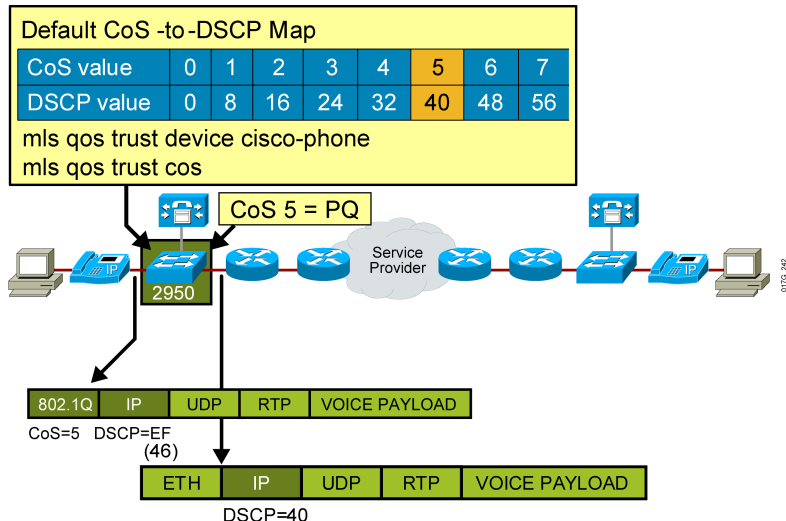
Note The IEEE 802.1Q standard is an IEEE specification for implementing VLANs in Layer 2 switched networks. IEEE 802.1Q and its use in QoS will be discussed further in the “Classification and Marking” module in this course.

With the frame marked at class of service (CoS) = 5 and differentiated services code point (DSCP) = Expedited Forwarding (EF), this frame should receive priority treatment every time it encounters any QoS mechanism in the network.

Note A User Datagram Protocol (UDP) header is used for voice packets rather than TCP.

Notice that a Real-Time Transport Protocol (RTP) header has been added because this is a voice packet. RTP helps synchronize real-time transmissions such as voice by time-stamping packets so that they can be resynchronized at the receiving end. This helps minimize jitter.

Life of a High-Priority VoIP Packet (Cont.)



In the Cisco Catalyst 2950 Switch, CoS = 5 means to treat the frame with priority queuing (PQ). This means that you should move the frame before any other frames with a lower CoS.

The default CoS-to-DSCP mapping is set to recognize the CoS = 5 as DSCP = 40. The EF value is 46 on input to the switch as set by the IP phone. Because the default CoS-to-DSCP marking is CoS 5 = DSCP 40 in the Cisco 2950 Switch (not 46), DSCP is set to 40 on output.

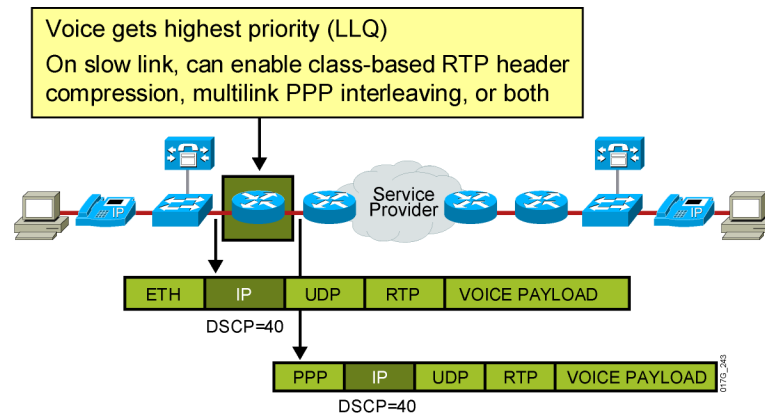
When the frame arrives at the Cisco Catalyst 2950 Switch, it is instantly recognized as a high-priority frame because of the CoS = 5 and is immediately placed in the high-priority, no-delay queue. Because the switch recognizes the frame as a CoS = 5, it re-marks the DSCP field to 40.

Default CoS-to-DSCP Mapping in Cisco Catalyst 2950 Switch

COS Value	DSCP Value
0	0
1	8
2	16
3	24
4	32
5	40
6	48
7	56

Note The default CoS-to-DSCP mappings are the same in Cisco Catalyst 2960 Switches and Cisco Catalyst 2950 Switches.

Life of a High-Priority VoIP Packet (Cont.)



When the packet hits the edge router, the router recognizes the packet as a voice packet due to the DSCP = 40 setting (which was set by the Cisco Catalyst 2950 Switch).

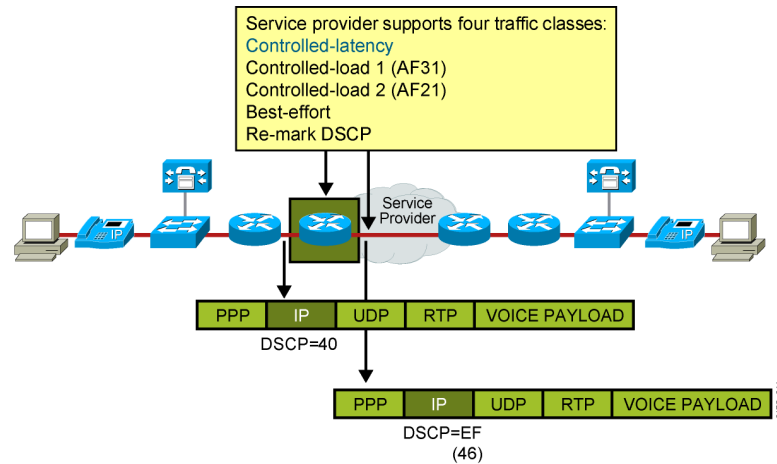
The packet is immediately dispatched ahead of any nonvoice packets, using low-latency queuing (LLQ). LLQ is designed to provide instant dispatch of voice packets ahead of data while carefully managing the dispatch of data.

If the link to the service provider is a relatively slow link, both header compression (in this case, class-based RTP header compression) and link fragmentation and interleaving (LFI) would be employed to improve the bandwidth efficiency of the link.

If the WAN link is a Frame Relay link, the packet would use Frame Relay traffic shaping (FRTS) and Frame Relay Forum 12 (FRF.12).

Note Both of these technologies are explained further in the “Traffic Shaping and Policing” module in this course.

Life of a High-Priority VoIP Packet (Cont.)



When the packet arrives at the service provider, the service provider reclassifies the packet to fit within the service provider QoS classification policy.

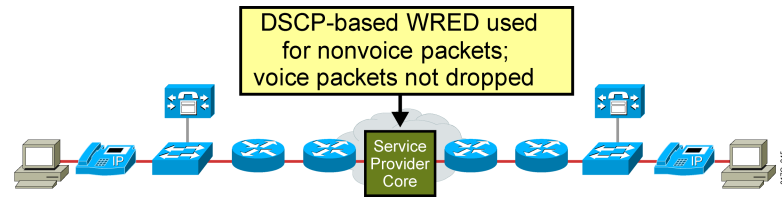
In this case, the service provider has defined these four traffic classes:

- Controlled latency
- Controlled load 1 (Assured Forwarding [AF]31)
- Controlled load 2 (AF21)
- Best-effort

The service provider provides QoS service level agreement (SLA) for the controlled-latency, controlled-load 1 (AF31), controlled-load 2 (AF21), and best-effort traffic classes. The service provider maps the enterprise customer QoS classifications into the four defined traffic classes of the service provider.

The service provider router recognizes the packet as a high-priority voice packet and assigns the packet to the real-time EF class. The packet is re-marked to DSCP = 46 to fit the service provider classification conventions and sent on its way as a member of the real-time class.

Life of a High-Priority VoIP Packet (Cont.)



© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-2.7

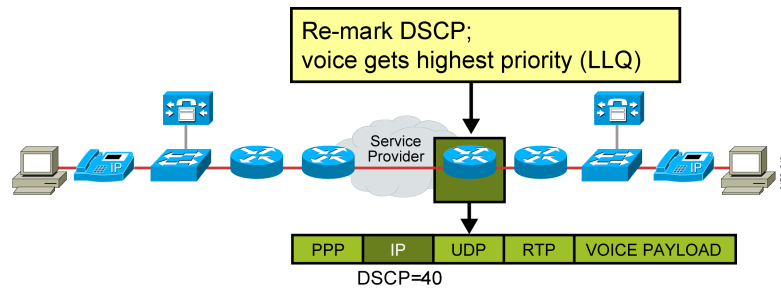
In the service provider core network, the packet will move along with minimal delay using EF.

The key congestion avoidance technology, weighted random early detection (WRED), is used in the service provider network. WRED ensures that lower-priority packets are dropped to ensure that priority packets make their way quickly through the network.

Because the voice packet is marked as EF—the service provider real-time class—WRED should have no impact on the packet. QoS policy for the service provider should be *not* to drop voice packets so that WRED would not be applied to packets identified as “real time.”

The packet will almost certainly not be dropped and will encounter minimal delay.

Life of a High-Priority VoIP Packet (Cont.)



© 2009 Cisco Systems, Inc. All rights reserved.

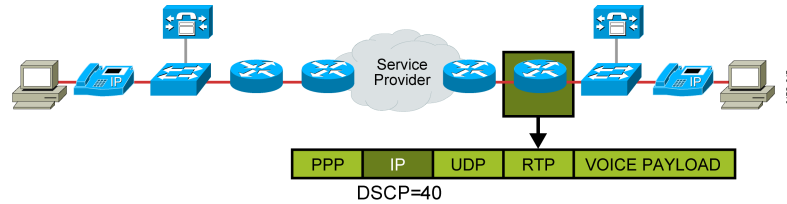
QoS v2.3-2.8

While the packet traversed the service provider network, the DSCP was marked as 46 so that the packet would be immediately dispatched as a member of the service provider real-time class.

But the packet was marked DSCP = 40 by the customer before it entered the service provider network. At the edge of the service provider network, the DSCP is re-marked to “40” to match the classification scheme being used by the enterprise customer.

The packet is dispatched immediately using the LLQ method that always provides absolute priority to voice packets.

Life of a High-Priority VoIP Packet (Cont.)

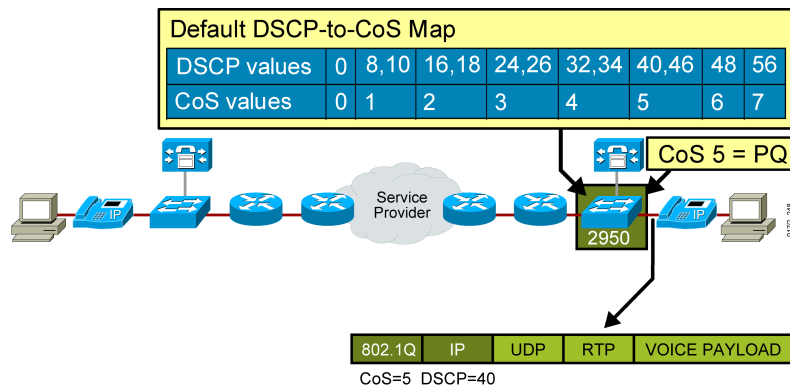


© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-2.8

Upon arriving at the enterprise network router, the voice packet is sent out the LAN interface toward the switch using FIFO queuing on the LAN interface.

Life of a High-Priority VoIP Packet (Cont.)

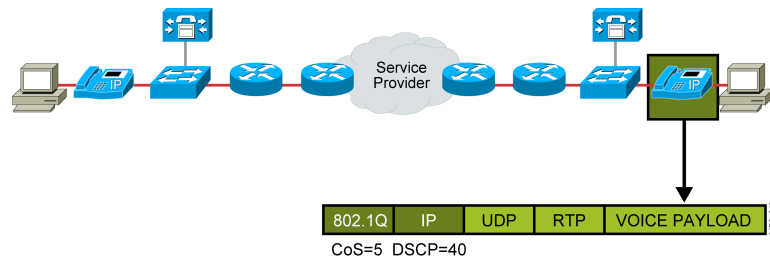


© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-2.10

Upon arriving at the Cisco Catalyst 2950 switch, the DSCP field is mapped to the CoS field of the outgoing Ethernet frame using the DSCP-to-CoS mapping. This results in CoS being set to 5. At the output interface, it is queued as a high-priority voice frame because the Layer 2 priority is set to CoS = 5. The frame jumps ahead of any nonvoice frame and is immediately dispatched to the PQ.

Life of a High-Priority VoIP Packet (Cont.)



© 2009 Cisco Systems, Inc. All rights reserved.

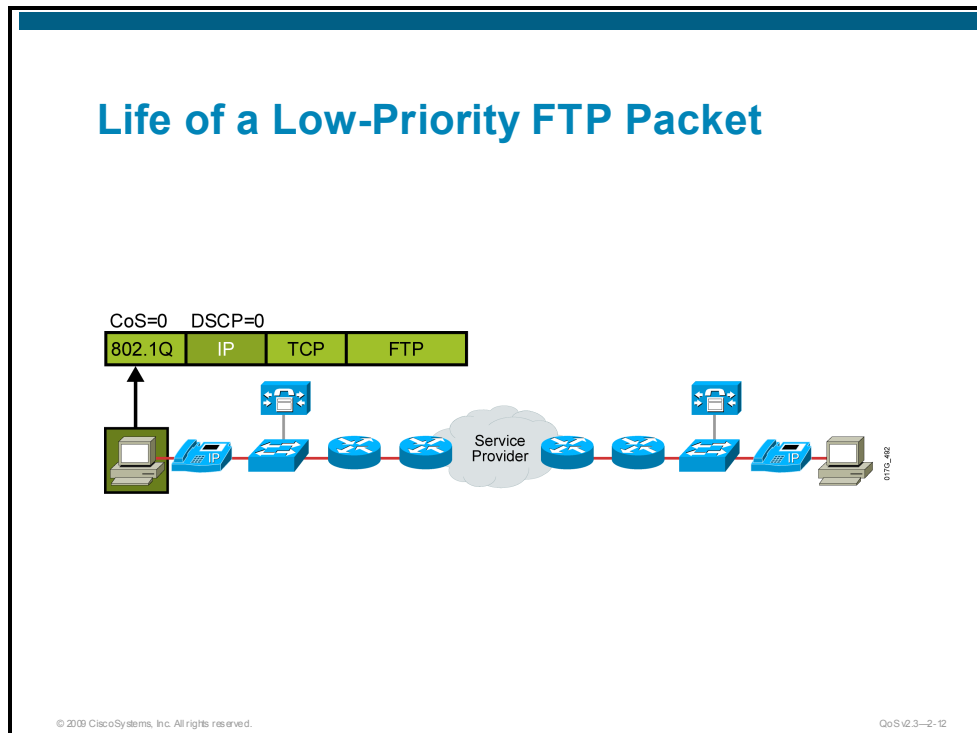
QoS v2.3—2-11

The packet finally arrives at the receiving IP phone.

The RTP header is used to ensure that the packet is synchronized correctly with other packets from the same voice flow and that the voice payload is delivered.

Life of a Low-Priority FTP Packet

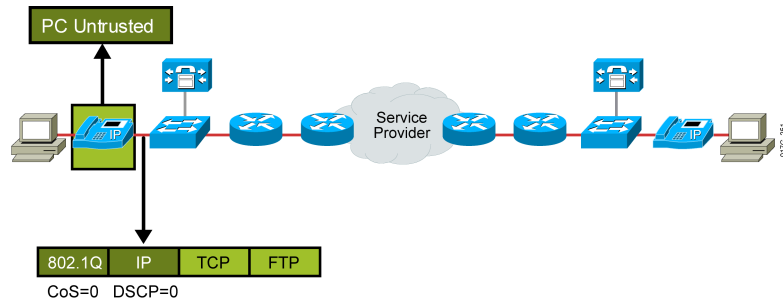
This topic describes the life of a low-priority FTP packet.



The low-priority FTP packet begins life as a very low-priority CoS = 0, DSCP = 0.

Note The FTP packet is using TCP rather than UDP (which was used by the voice packet).

Life of a Low-Priority FTP Packet (Cont.)



© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-2-13

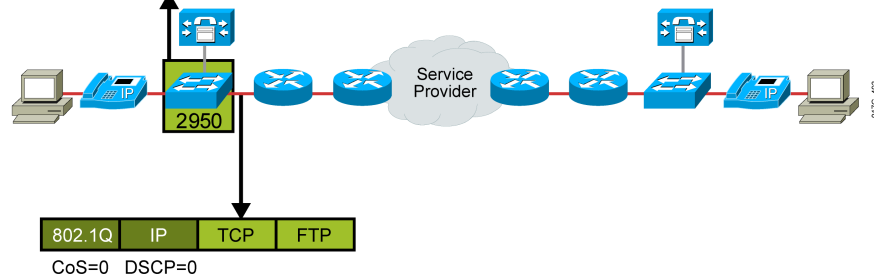
All traffic arriving from the workstation attached to the IP phone is set to be untrusted.

As a result, the IP phone will not accept any CoS marking done by the workstation and will re-mark the CoS to 0. This ensures that the voice traffic generated by the IP phone will always receive priority treatment over any traffic generated by the workstation.

Note The IP phone cannot re-mark the incoming DSCP value. The IP phone can only re-mark the Layer 2 CoS value.

Life of a Low-Priority FTP Packet (Cont.)

- Default CoS-to-DSCP mapping, CoS 0 = DSCP 0
- CoS 0 will use one of the WRR queues.
- mls qos trust device cisco-phone
- mls qos trust cos



In the Cisco Catalyst 2950 switch, the CoS-to-DSCP mapping table is used to map the CoS value of the packet to the DSCP equivalent. In the case of the FTP frame, the DSCP = 0 matches the CoS = 0, so the frame DSCP value does not change.

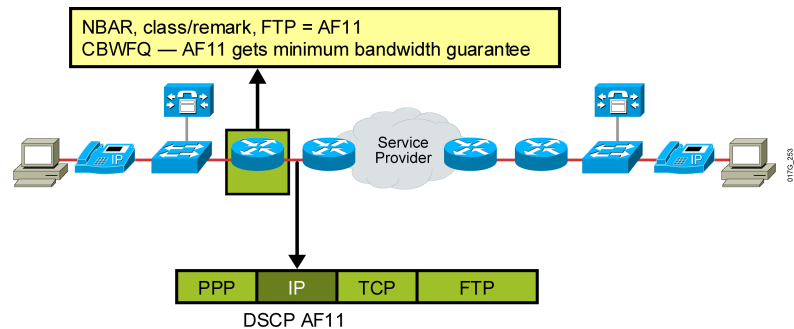
The switch congestion management technology—weighted round robin (WRR) with an expedite queue (priority queue [PQ])—dispatches the frame, but not until all high-priority voice frames have been dispatched.

Note WRR with an expedite queue is explained further in the “Congestion Management” module in this course.

Default CoS-to-DSCP Mapping in Cisco Catalyst 2950 Switch

COS Value	DSCP Value
0	0
1	8
2	16
3	24
4	32
5	40
6	48
7	56

Life of a Low-Priority FTP Packet (Cont.)



© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3—2-15

In the enterprise router, a classification technology, Network-Based Application Recognition (NBAR), recognizes the packet as an FTP packet and re-marks the packet with a DSCP = “001010” = AF11.

With a DSCP = AF11, the packet is then dispatched as a low-priority class packet by CBWFQ. CBWFQ is the component of LLQ that carefully manages the dispatch of data traffic.

The AF11 class is given a minimum guarantee of bandwidth. If the link to the service provider is congested, the packet has a good probability of being dropped to ensure that higher-priority packets are not delayed.

Life of a Low-Priority FTP Packet (Cont.)

The diagram illustrates the path of a low-priority FTP packet through a service provider network. At the top, a yellow box lists the service provider's traffic classes: Controlled-latency, Controlled-load 1 (AF31), Controlled-load 2 (AF21), Best-effort, and Re-mark DSCP. The network topology shows a sequence of devices: a PC, a blue IP phone, a blue switch, a blue router, a green switch (the service provider's edge router), a cloud labeled 'Service Provider', another green switch, a blue router, a blue switch, another blue IP phone, and a final PC. A red line represents the network path. A packet, represented by a blue square with a white 'X', is shown entering the network from the left. It passes through the first blue switch and router. At the green switch, it is classified into 'Controlled-load 2 (AF21)'. Below this switch, a green box shows the packet's header fields: PPP, IP, TCP, and FTP. The DSCP value is marked as 'AF11'. The packet then enters the 'Service Provider' cloud. After exiting the cloud, it passes through another green switch, where it is re-marked with 'DSCP AF21'. Below this switch, another green box shows the header fields: PPP, IP, TCP, and FTP. The packet then passes through the final blue router and switch before reaching the destination PC. A yellow box on the left states 'FTP = Controlled-load 2 (AF21)'. The diagram is labeled '017C_04' in the bottom right corner.

Service provider supports four traffic classes:
Controlled-latency
Controlled-load 1 (AF31)
Controlled-load 2 (AF21)
Best-effort
Re-mark DSCP

Service Provider

FTP = Controlled-load 2 (AF21)

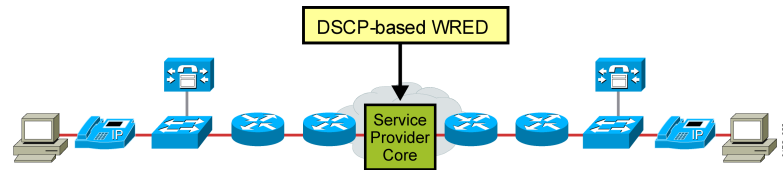
DSCP AF11

DSCP AF21

017C_04

Upon arriving at the service provider network, the packet is identified as an FTP packet and assigned to the controlled-load 2 (AF21) class.

Life of a Low-Priority FTP Packet (Cont.)



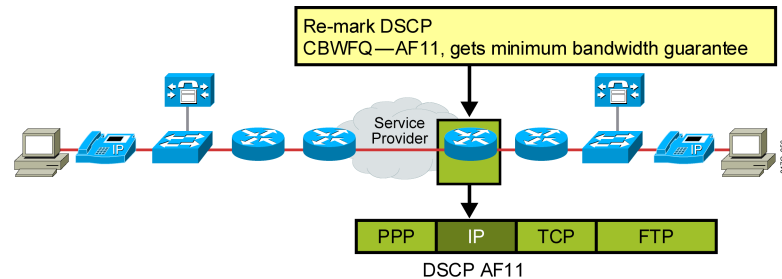
© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-2-17

The packet traverses the service provider core marked as a controlled-load 2 (AF21) packet.

While in the service provider core network, the FTP packet has a much better probability of being dropped by WRED than the voice packet.

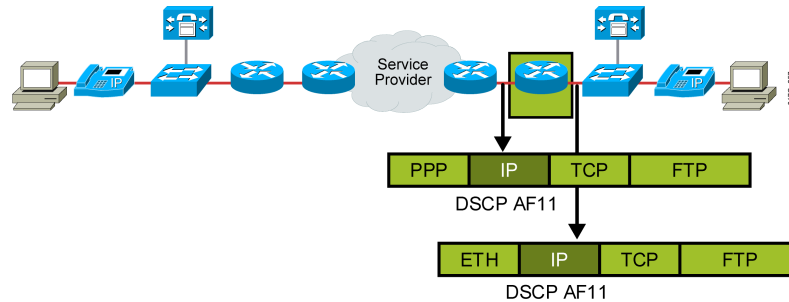
Life of a Low-Priority FTP Packet (Cont.)



Before the packet entered the service provider network, it was marked DSCP = AF11, which fits the classification scheme used by the enterprise customer. As the packet leaves the service provider network, the packet is re-marked to DSCP = AF11 for the enterprise customer.

The AF11 class is given a minimum guarantee of bandwidth.

Life of a Low-Priority FTP Packet (Cont.)



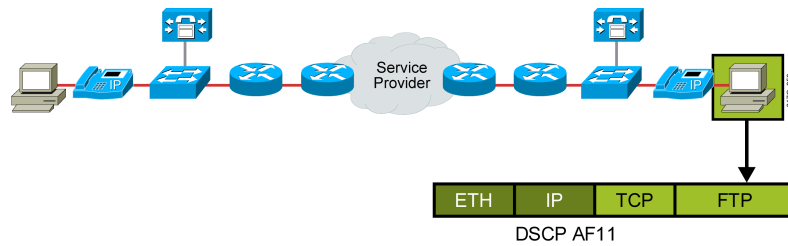
© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-2-19

As the packet reenters the enterprise network, it is recognized as an AF11 class packet and is passed through the enterprise router without being re-marked.

The FTP packet is sent out the LAN interface toward the switch using FIFO queuing on the LAN interface.

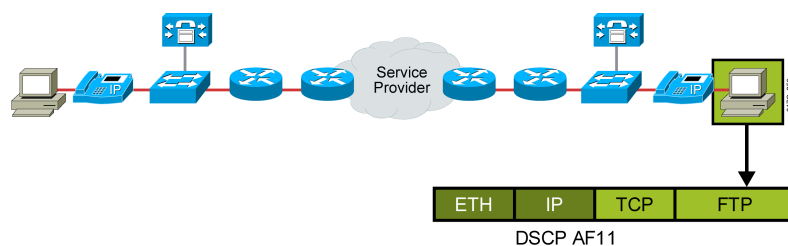
Life of a Low-Priority FTP Packet (Cont.)



Using its DSCP-to-CoS mapping, the Cisco Catalyst 2950 Switch recognizes the DSCP = AF11 packet (Layer 3) as a CoS = 1 priority frame (Layer 2).

The FTP frame is treated by WRR with the CoS 1.

Life of a Low-Priority FTP Packet (Cont.)



The FTP packet finally arrives at the destination host, and the payload is delivered.

If the packet had been dropped at any point along the way, TCP would recognize that fact and request retransmission of the packet.

Summary

This topic summarizes the key points discussed in this lesson.

Summary

- High-priority and low-priority packets are treated very differently in a network using differentiated services.
- The high-priority VoIP packet begins life at an IP phone as a CoS 5 on the LAN, which translates to DSCP 40 as the packet hits the WAN and is given EF status in the service provider core network; the high-priority packet is immediately transmitted by all devices as it moves through the network.
- The low-priority FTP packet begins life as a CoS 0 packet, which translates to DSCP 0 and AF11 in the service provider core network.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3—2-22

Module Summary

This topic summarizes the key points discussed in this module.

Module Summary

- Classification is a critical QoS component that recognizes and distinguishes between different traffic streams. Without classification, all packets are treated the same.
- Marking is a QoS component that “colors” a packet so that it can be identified and distinguished from other packets in QoS treatment.
- MQC classification can be achieved using class maps and a variety of traffic descriptors, including: CoS (ISL, 802.1Q), IP precedence, DSCP, QoS group, MPLS experimental bits, Frame Relay DE bit, and ATM CLP bit.
- Many different mechanisms exist to perform classification and marking, including: class maps, class-based marking, NBAR, QoS preclassify, QPPB, and LAN-based CoS marking.
- As packets travel through the network, each packet is classified and marked as close to the source as possible. The downstream devices then apply different QoS mechanisms to the packet based on its marking. The packets can be re-marked along the way.

© 2009 Cisco Systems, Inc. All rights reserved.

QoS v2.3-4.1

Classification is the process of identifying traffic and categorizing it into different classes. Packet classification allows some packets to be handled more quickly or with a higher priority than other packets. Applications such as voice typically need to be treated faster than a file transfer.

Classification uses a traffic descriptor to categorize a packet within a specific group, to define that packet. Typically, traffic descriptors include: Class of service (CoS) (Inter-Switch Link [ISL], 802.1Q) incoming interface, IP precedence, differentiated services code point (DSCP), QoS group ID, Multiprotocol Label Switching (MPLS) experimental bits, Frame Relay discard eligible (DE) bit, ATM cell loss priority (CLP) bit, source or destination address, or application.

Marking a packet or frame with its classification allows network devices to easily distinguish the marked packet or frame. Marking is a useful feature in that it allows network devices to easily identify packets or frames as belonging to a specific class. After packets have been identified as belonging to a specific class, QoS mechanisms can be uniformly applied to ensure compliance with administrative QoS policies.

You can implement packet classification using such tools as Modular quality of service (QoS) command-line interface (CLI), or MQC, class maps and policy maps, Network-Based Application Recognition (NBAR), QoS preclassify (VPN QoS), and QoS Policy Propagation via BGP (QPPB).

Classification and marking can be performed at the network or link layer.

References

For additional information, refer to these resources:

- For an overview of classification, refer to *Classification Overview* at this URL:
http://www.cisco.com/en/US/docs/ios/qos/configuration/guide/classification_oview_ps6350_TSD_Products_Configuration_Guide_Chapter.html
- For additional information on 802.1p/Q marking, refer to *Bridging Between IEEE 802.1Q VLANs* at this URL:
http://www.cisco.com/en/US/docs/ios/12_1/12_1t3/feature/guide/dtbridge.html
- For additional information on ISL marking, refer to *Configuring Routing between VLANs with ISL Encapsulation* at this URL:
http://www.cisco.com/en/US/docs/ios/12_2/switch/configuration/guide/xcfvliisl.html
- For additional information on ISL marking, refer to *Configuring QoS* at this URL:
<http://www.cisco.com/en/US/docs/switches/lan/catalyst6500/catos/8.x/configuration/guide/qos.html>
- For more information on classification using MQC, refer to *Configuring the Modular Quality of Service Command-Line Interface* at this URL:
http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fqos_c/fqcprt8/qcfmcli2.pdf
- For more information on class-based marking, refer to *Class-Based Marking* at this URL:
<http://www.cisco.com/univercd/cc/td/doc/product/software/ios121/121newft/121t/121t5/cbpmark2.pdf>
- For more information on NBAR, refer to *Network-Based Application Recognition* at this URL:
http://www.cisco.com/en/US/prod/collateral/iosswrel/ps6537/ps6558/ps6612/ps6653/prod_qas09186a00800a3ded.html
- For more information on QoS preclassify, refer to *Reference Guide to Implementing Crypto and QoS* at this URL:
http://www.cisco.com/en/US/tech/tk543/tk757/technologies_tech_note09186a00800b3d15.shtml
- For more information on QoS preclassify, refer to *Configuring QoS for Virtual Private Networks* at this URL:
http://www.cisco.com/en/US/docs/ios/12_2/qos/configuration/guide/qcfvpn.html
- For more information on QPPB, refer to “*Quality of Service Policy Propagation via BGP*” at the following URL:
http://www.cisco.com/en/US/docs/ios/11_1/feature/guide/bgppprop.html
- For more information on configuring QPPB, refer to “*Configuring QoS Policy Propagation via Border Gateway Protocol*” at the following URL:
http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fqos_c/fqcprt1/qcfprop.pdf
- For more information on classification and marking on the Catalyst 2960, refer to “*Configuring QoS*” at the following URL:
http://www.cisco.com/en/US/docs/switches/lan/catalyst2960/software/release/12.2_46_se/configuration/guide/swqos.html

- For more information on classification and marking on the Catalyst 3750, refer to “*Configuring QoS*” at the following URL:
http://www.cisco.com/en/US/docs/switches/lan/catalyst3750/software/release/12.2_46_se/configuration/guide/swqos.html
- For more information on classification and marking on the Catalyst 6500, refer to “*Configuring QoS*” at the following URL:
<http://www.cisco.com/en/US/docs/switches/lan/catalyst6500/catos/8.x/configuration/guide/qos.html>

Module Self-Check

Use the questions here to review what you learned in this module. The correct answers and solutions are found in the Module Self-Check Answer Key.

- Q1) Classification of packets should occur _____. (Source: Understanding Classification and Marking)
- A) at the distribution layer
 - B) anywhere in the core of the network
 - C) as close to the source of the traffic as possible
 - D) as close to the destination of the traffic as possible
- Q2) To use a class map, QoS must be referenced through the use of _____. (Source: Using MQC for Classification)
- A) a route map
 - B) an access list
 - C) a policy map
 - D) a service map
- Q3) What is a requirement for using class-based marking? (Source: Using MQC for Class-Based Marking)
- A) Cisco Express Forwarding must be enabled.
 - B) Cisco Express Forwarding must be disabled.
 - C) Cisco Express Forwarding can only be used on serial interfaces.
 - D) Cisco Express Forwarding can only be used on Ethernet interfaces.
- Q4) What is the MQC feature that allows traffic to be classified by a packet subport value? (Source: Using NBAR for Classification)
- A) LDPM
 - B) NBAR
 - C) service maps
 - D) service classes
- Q5) The QoS for VPNs feature is designed to operate on _____. (Source: Configuring QoS Preclassify)
- A) logical interfaces
 - B) loopback interfaces
 - C) tunnel interfaces
 - D) physical interfaces
- Q6) Which of these commands will modify the IP precedence or QoS group value when the IP routing table is updated with BGP learned routes? (Source: Configuring QoS Policy Propagation via BGP)
- A) **table-map**
 - B) **bgp-policy**
 - C) **map bpg ip**
 - D) **bgp table-map**

- Q7) Which of these commands will enable the propagation of the QoS policy via BGP on an interface? (Source: Configuring QoS Policy Propagation via BGP)
- A) **table-map**
 - B) **bgp-policy**
 - C) **bgp send-policy**
 - D) **bgp policy-propagation**
- Q8) Which of these commands will display both the CoS-to-DSCP and DSCP-to-CoS mappings on a Catalyst switch? (Source: Configuring LAN Classification and Marking)
- A) **show mls maps**
 - B) **show mls qos maps**
 - C) **show mls maps both**
 - D) **show qos mls maps both**

Module Self-Check Answer Key

Q1)	C
Q2)	C
Q3)	A
Q4)	B
Q5)	C
Q6)	A
Q7)	B
Q8)	B

