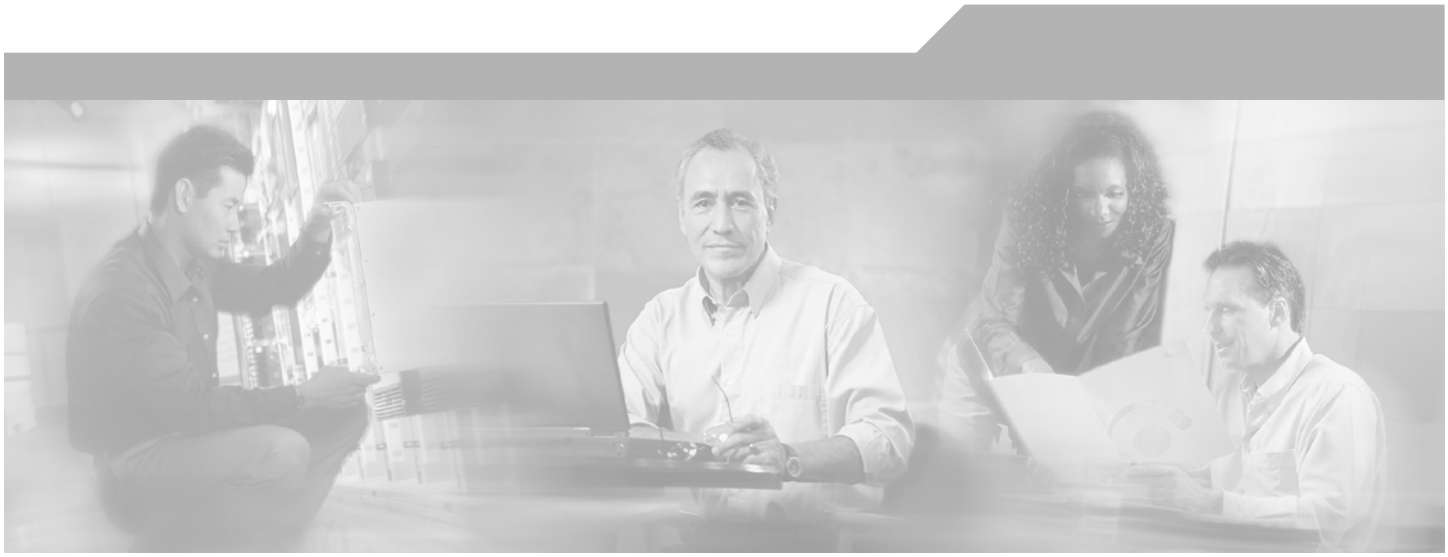
Cisco Systems



Next Generation Enterprise MPLS VPN-Based MAN Design and Implementation Guide

OL-11661-01

Corporate Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 526-4100

CONTENTS

CHAPTER 1 **Problems/Solution Description** 1-1

Deploying VPNs 1-3

CHAPTER 2 **Technology Overview** 2-1

MPLS 2-1

MPLS Layer 3 VPNs 2-1

Multipath Load Balancing 2-3

OSPF as the PE-CE Routing Protocol 2-4

OSPF and Backdoor Links 2-4

EIGRP as PE-CE Routing Protocol 2-6

EIGRP and Backdoor Links 2-6

MPLS Network Convergence 2-8

Site-to-Site VPN Convergence Description with Default Timers 2-8

MPLS Network Convergence Tuning Parameters 2-9

EIGRP 2-9

OSPF 2-10

BGP 2-10

LDP 2-10

Bidirectional Forwarding Detection (BFD) 2-10

Scalability of an MPLS Network 2-11

MPLS Layer 2 VPNs—AToM 2-11

Ethernet over MPLS 2-12

QoS in AToM 2-13

Scalability 2-13

EoMPLS Sample Configuration 2-13

CHAPTER 3 **MPLS-Based VPN MAN Reference Topology** 3-1

MAN Topology 3-1

VPN Information 3-2

Inventory of Devices 3-3

Building a MAN MPLS VPN Network 3-4

CHAPTER 4 **Implementing Advanced Features on MPLS-Based VPNs** 4-1

QoS for Critical Applications 4-1

QoS Design Overview 4-1
Strategically Defining the Business Objectives 4-2
Analyzing the Service Level Requirements 4-4
QoS Requirements of VoIP 4-4
QoS Requirements of Video 4-5
QoS Requirements of Data 4-7
QoS Requirements of the Control Plane 4-9
Scavenger Class QoS 4-10
Designing the QoS Policies 4-10
QoS Design Best Practices 4-10
NG-WAN/MAN QoS Design Considerations 4-12
MPLS DiffServ Tunneling Modes 4-12
Security 4-15
Encryption 4-15
VPN Perimeter—Common Services and the Internet 4-16
Unprotected Services 4-19
Firewalling for Common Services 4-19
Network Address Translation—NAT 4-21
Common Services 4-22
Single Common Services—Internet Edge Site 4-22
Multiple Common Services—Internet Edge Sites 4-24
Internet Edge Site Considerations 4-25
Routing Considerations 4-28
NAT in the MPLS MAN 4-29
Convergence 4-31
Traffic Engineering Fast ReRoute (TE FRR) 4-31
Fast Reroute Activation 4-32
Backup Tunnel Selection Procedure 4-33
Protecting the Core Links 4-34
Performance 4-35

CHAPTER 5 **Management** 5-1
Related Documents 5-2

CHAPTER 6 **Advanced Applications Over MPLS-Based VPNs** 6-1
Cisco IP Communications 6-1
Overview of Cisco IP Communications Solutions 6-1
Overview of the Cisco IP Telephony Solution Over the Self-Managed MPLS MAN 6-2
Cisco IP Network Infrastructure 6-5

Quality of Service 6-5
Call Processing Agent 6-5
Communication Endpoints 6-6
Applications 6-6
IP Telephony Deployment Models over the Self-Managed MPLS MAN 6-8
Multi-Site MPLS MAN Model with Distributed Call Processing 6-8
Clustering over the MPLS MAN 6-10
Intra-Cluster Communications 6-12
Failover between Subscriber Servers 6-12
Cisco CallManager Publisher 6-12
Call Detail Records (CDR) 6-13
Multi-Site MPLS MAN Model with Centralized Call Processing 6-17
Survivable Remote Site Telephony 6-21
Network Infrastructure 6-23
Campus Access Layer 6-24
CallManager Server Farm 6-26
Network Services 6-26
Media Resources 6-27
Music on Hold 6-27
Deployment Basics of MoH 6-27
Unicast and Multicast MoH 6-28
Recommended Unicast/Multicast Gateways 6-28
MoH and QoS 6-29
Call Processing 6-29
Cisco Unity Messaging Design 6-29
Messaging Deployment Models 6-29
Messaging Failover 6-30
Multicast 6-31
Multicast VPN Service Overview 6-32
Multicast VPN Service Architecture 6-32
Service Components 6-32
Multiprotocol BGP 6-33
New Extended Community Attribute 6-33
MVRF 6-34
Multicast Tunnel Interface (MTI) 6-34
Multicast Domain (MD) 6-34
Multicast Distribution Tree (MDT) 6-34
Multicast VPN Service Design and Deployment Guidelines 6-34
Service Deployment 6-35
Multicast Core Configuration—Default and Data MDT Options 6-37

Caveats 6-38
QoS for mVPN Service 6-39
Multicast VPN Security 6-40
Design Choices for Implementing mVPN 6-45
Implementing and Configuring the mVPN Service 6-46
Ethernet over MPLS 6-50
EoMPLS Overview 6-50
EoMPLS Architecture 6-51
MPLS VC Circuit Setup 6-52
Technical Requirements for EoMPLS 6-53
EoMPLS Restrictions 6-55
Configuration and Monitoring 6-56
PXF-Based Cisco 7600 Configuration 6-56
Cisco 12K Configuration 6-56
Cisco 7200 Configuration 6-56
Cisco 3750 Metro Configuration 6-57
Cisco PXF-Based and Cisco 12K Monitoring Commands 6-57
Cisco PFC-Based Configuration 6-58
Cisco PXF-Based Monitoring Commands 6-58

CHAPTER 7 **MPLS-Based VPN MAN Testing and Validation** 7-1
Test Topology 7-1
Test Plan 7-5
Baseline MPLS VPN 7-5
Security 7-5
QoS 7-6
Data 7-6
Voice 7-6
Multicast 7-8
MPLS Network Convergence 7-9
Convergence Test Results 7-10
Core IGP—EIGRP 7-11
Core Protocol—OSPF 7-13

CHAPTER 8 **Configurations and Logs for Each VPN** 8-1
Cisco 7600 as PE 8-1
Cisco 12000 as PE 8-7
Service Validation 8-14
Core Verification 8-14

Edge Verification 8-17
Baseline MPLS VPN 8-17
OSPF Backdoor Link Verifications 8-20
QoS 8-24
Multicast 8-26

APPENDIX A **Platform-Specific Capabilities and Constraints** A-1

Cisco 7200 QoS Design A-1
Cisco 7200—Uniform Mode MPLS DiffServ Tunneling A-1
Cisco 7200—8-Class QoS Model A-2
Cisco 7200—11-Class QoS Model A-5
Cisco 7304 QoS Design A-7
Classification A-8
Policing A-8
Weighted Random Eary Detection (WRED) A-9
Class-based Weighted Fair Queuing (CBWFQ) A-10
Hierarchical Policies A-10
Cisco 7600 QoS Design A-13
Cisco 7600—Uniform Mode MPLS DiffServ Tunneling A-13
Cisco 7600—Trust States and Internal DSCP Generation A-13
Cisco 7600—Queuing Design A-16
Cisco 7600 1P2Q1T 10GE Queuing Design A-18
Cisco 7600 1P2Q2T GE Queuing Design A-19
Cisco 7600 1P3Q1T GE Queuing Design A-21
Cisco 7600 1P3Q8T GE Queuing Design A-23
Cisco 7600 1P7Q8T 10GE Queuing Design A-25
Cisco 12000 QoS Design A-28
Cisco 12000 GSR Edge Configuration A-28
PE Config (CE Facing Configuration—Ingress QoS) A-30
PE Config (CE Facing Configuration—Egress QoS) A-30
PE Config (P Facing Configuration—Ingress QoS) A-33
Cisco 12000 GSR ToFab Queuing A-34
WRED Tuning at the Edge and Core A-35

APPENDIX B **Terminology** B-1

CHAPTER 1

Problems/Solution Description

Cisco enterprise customers have in the past relied heavily upon traditional WAN/MAN services for their connectivity requirements. Layer 2 circuits based on TDM, Frame Relay, ATM, and SONET have formed the mainstay of most low-speed WAN services. More recently, high-speed MAN solutions have been delivered directly over Layer 1 optical circuits, SONET, or through the implementation of point-to-point or point-to-multipoint Ethernet services delivered over one of these two technologies.

Today, many enterprise customers are turning to Multiprotocol Label Switching (MPLS)-based VPN solutions because they offer numerous secure alternatives to the traditional WAN/MAN connectivity offerings. The significant advantages of MPLS-based VPNs over traditional WAN/MAN services include the following:

- Provisioning flexibility
- Wide geographical availability
- Little or no distance sensitivity in pricing
- The ability to mix and match access speeds and technologies
- Perhaps most importantly, the ability to securely segment multiple organizations, services, and applications while operating a single MPLS-based network

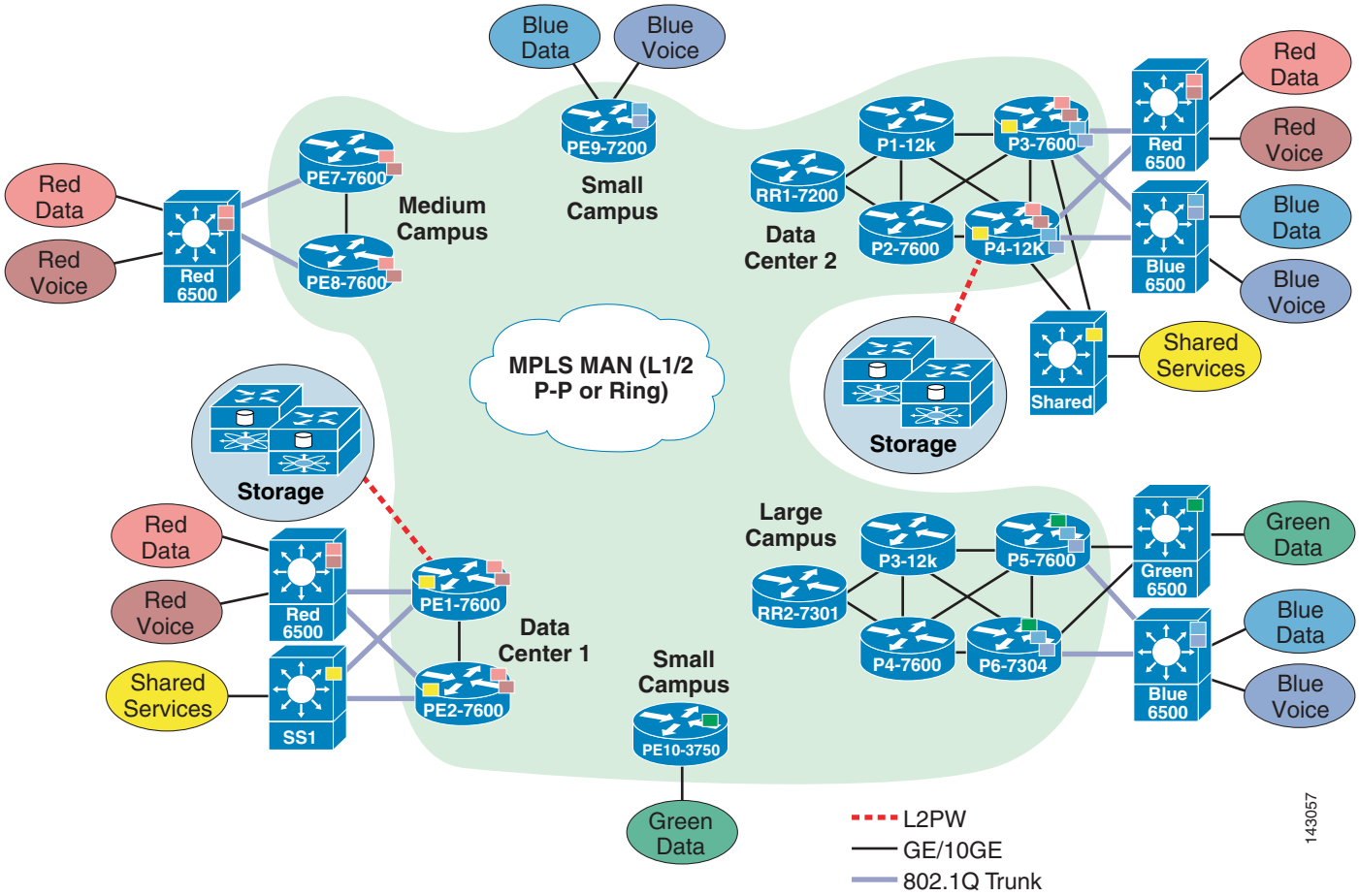
Although service providers have been offering managed MPLS-based VPN solutions for years, the largest enterprise customers are now beginning to investigate and deploy MPLS in their own networks to implement self-managed MPLS-based VPN services. The concept of self-managed enterprise networks is not new; many enterprise customers purchase Layer 2 TDM, Frame Relay, or ATM circuits and deploy their own routed network for these circuits. The largest of enterprise customers even manage their own core networks by implementing Frame Relay or ATM-based switching infrastructures and "selling" connectivity services to other organizations within their companies.

Both of these solutions have had disadvantages; deploying an IP-based infrastructure over leased lines offers little flexibility and segmentation capabilities that are cumbersome at best. Deploying a switched Frame Relay or ATM infrastructure to allow for resiliency and segmentation is a solution within reach of only the largest and most technically savvy enterprises.

As noted, the self-managed MPLS-based network is typically reserved for larger enterprises willing to make a significant investment in network equipment and training, with an IT staff that is comfortable with a high degree of technical complexity. A self-managed MPLS VPN can be an attractive option if a business meets these requirements and wants to fully control its own WAN or MAN and to increase segmentation across multiple sites to guarantee delivery of specific applications. The level of security between separated networks is comparable to private connectivity without needing service provider intervention, allowing for consistent network segmentation of departments, business functions, and user groups.

Corporations with a propensity for mergers and acquisitions benefit from the inherent any-to-any functions of MPLS that, when the initial configuration is completed, allow even new sites with existing networks to be merged with the greater enterprise network with minimal overhead. Secure partner networks can also be established to share data and applications as needed, on a limited basis. The self-managed MPLS is also earning greater adoption as an important and viable method for meeting and maintaining compliance with regulatory privacy standards such as HIPAA and the Sarbanes-Oxley Act. A typical description of this model is "an enterprise acting as a service provider." Figure 1-1 shows a typical self-managed MPLS MAN deployment.

Figure 1-1 Typical Self-Managed MPLS MAN Deployment



The following chapters of this guide:

- Explore the technologies necessary to implement a self-managed MPLS-based VPN.
- Describe the evolution of a traditional IP-based enterprise network into an MPLS-based segmented MAN network.
- Discuss the implementation of advanced features such as high availability, QoS, security, and common network services such as NAT, DNS, DHCP, and messaging.
- Explore the management of MPLS VPNs.
- Describe key MPLS-based VPN services such as multicast VPNs and Ethernet over MPLS pseudowires.

- Describe the test bed, test scenarios, and configuration guidelines/caveats associated with recent testing of the MPLS-based VPN MAN topology based on the reference topology described in Figure 1-1.

Deploying VPNs

While the technology enables you to create the logical separation across networks, it is important to understand the reasons for creating these logical networks. Enterprise customers increasingly require segmentation for a number of different reasons:

- Closed User Groups (CUG)—The CUGs could be created based on a number of different business criterias, with guest Internet access for onsite personnel being the simplest example. Providing NAC/isolation services also creates a need to separate the non-conforming clients. While this can be done using VLANs within a Layer 2 campus network, it requires Layer 3 VPN functionality to extend it across Layer 3 boundaries. CUGs could be created with partners, either individually or as a sub-group, where the segmentation criteria are resources that are to be shared/accessed. This simplifies the information sharing with partners while still providing security and traffic separation.
- Virtualization—Segmentation to the desktop is driving virtualization in the application server space. This means that even existing employees can be segmented into different CUGs where they are provided access to internal services based on their group membership.
- Enterprise as a Service Provider—With some of the Enterprise networks expanding as their organization expands, IT departments at some of the large Enterprises have become internal Service Providers. They leverage a shared network infrastructure to provide network services to individual Business Units within the Enterprise. This not only requires creating VPNs, but also requires the ability of each of the BUs to access shared corporate applications. Such a model can be expanded to include scenarios in which a company acquires another company (possibly with an overlapping IP addressing scheme) and needs to eventually consolidate the networks, the applications, and the back office operations.
- Protecting critical applications—Another segmentation criteria could be based off the applications themselves rather than the users. An organizations that feels that its critical applications need to be separated from everyday network users can create VPNs for each or a group of applications. This not only allows it to protect them from any malicious traffic, but also more easily control user access to the applications. An example of this is creating separate VPNs for voice and data.

Beyond the segmentation criteria, the overarching considerations should be based on the need to share. The VPNs create a closed user group that can easily share information but there will always be the scenario that requires sharing across the VPNs. For example, a company-wide multicast stream would need to be accessible by all the employees irrespective of their group association. Thus the VPNs should be created based on practical considerations that conform to the business needs of the organization.

CHAPTER 2

Technology Overview

MPLS

MPLS was viewed until recently as a service provider routing technology. Next generation enterprise networks relying on intelligent network infrastructure for solutions such as IP telephony, storage, wireless, and the applications and services that surround them demand network resiliency and features no less than and at times exceeding what is available in service provider networks. Using MPLS to build VPNs in enterprise networks addresses new requirements such as network segmentation, extending segmentation across campuses, address transparency, and shared services in the most scalable way while leveraging the benefits and flexibility of IP. MPLS application components include Layer 3 VPNs, Layer 2 VPNs, QoS, and Traffic Engineering. The following sections focus on Layer 3 and Layer 2 VPNs as these are the key applications for Enterprise networks.

MPLS Layer 3 VPNs

The following components perform a specific role in successfully building an MPLS VPN network:

- Interior Gateway Protocol (IGP)—This routing protocol is used in an MPLS core to learn about internal routes.

  The IGP table is the global routing table that includes routes to the provider edge (PE) routers or any provider (P) router. Note that these routes are not redistributed into the VPN (external site) routes.

  Although any routing protocol including static routes can be used in the MPLS core, using a dynamic routing protocol such as EIGRP or OSPF that gives sub-second convergence is more desirable. If the customer is required to support MPLS Traffic Engineering applications, then a link-state protocol such as OSPF or IS-IS is required.

- Cisco Express Forwarding table—Derived from FIB and LFIB tables and used to forward VPN traffic.

- Label Distribution Protocol (LDP)—Tag Distribution Protocol (TDP) is the precursor to LDP and was invented by Cisco Systems. TDP is a proprietary protocol. TDP and LDP use the same label format but the message format is different.

  LDP supports the following features that are not part of TDP:

  - Extension mechanism for vendor-private and experimental features
  - Backoff procedures for session establishment failures
  - Abort mechanism for LSP setup
  - Optional use of TCP MD5 signature option for (more) secure operation

- Optional path-vector-based loop detection mechanism to prevent setup of looping LSPs
- More combinations of modes of operation

It is recommended to use LDP when possible.

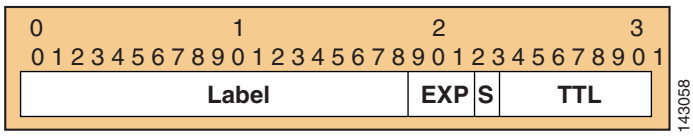
LDP is enabled on the core routers (P) and the PE core-facing interfaces to generate and distribute labels appropriately for the prefixes that were learned from the core IGP.

After the IGP in the core converges, labels are generated and bound to these prefixes and kept in the table that is referenced when the packets are forwarded. This table is called the Label Forwarding Information Base (LFIB).

Note that packets are switched based on pre-calculated labels and not routed through an MPLS core. Ingress Edge LSR (PE) appends a label before forwarding a packet to its neighbor LSR(P). The neighbor LSR swaps incoming label with outgoing label and forwards the packet to its neighbor. If this neighbor is Egress Edge LSR(PE), the core LSR(P) pops the label to avoid double look up (MPLS and IP) at the Egress Edge LSR and forwards the packet as an IP packet. This action of removing the label one hop prior to reaching the egress LSR is called Penultimate Hop Popping (PHP).

- MPLS labels and label stacking—The MPLS label is 32 bits and is used as a shim header in the forwarding plane (See Figure 2-1). Each application is responsible for generating labels. Labels generated in the control plane are used in the forwarding plane and encapsulated between Layer 2 and Layer 3 packet headers.

Figure 2-1 MPLS Labels



- Label = 20 bits
- CoS/EXP = Class of Service, 3 bits
- S = Bottom of stack, 1 bit
- TTL = Time to live, 8 bits

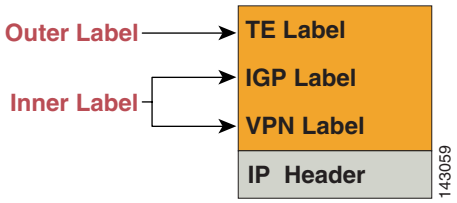
Label stacking occurs if more than one MPLS application is present in the network. For example:

- IGP labels (outer label if TE is not in use)—Used for forwarding packets in the core
- BGP labels (inner label)—Used for assigning end users/communities
- RSVP labels (outer label)—Used for TE tunnels

If TE, VPN, and MPLS are enabled, the headend and the tailend LSR are in charge of applying and removing TE label. If not, IGP is the outer most label and since PHP is on by default, an LSR one hop prior to the Egress Edge LSR removes this outer label. For VPN traffic, Ingress Edge LSR appends VPN label and Egress Edge LSR removes VPN label.

MPLS label stacking in Figure 2-2 demonstrates label stacking when multiple MPLS services are enabled in a network.

Figure 2-2 MPLS Label Stacking Example



- CE-PE routing protocols—Distribute VPN site routing information to a PE that is adjacently connected to a VPN site. Any of the EGP and IGP routing protocols including static, RipV2, EIGRP, OSPF, IS-IS, or eBGP are supported.
- Route distinguisher (RD)—A PE acquires knowledge about routes for multiple VPNs through a single BGP process. This process builds a BGP table containing prefixes belonging to VPNs that can possibly have overlapping address spaces. To enforce uniqueness of the prefixes held in the BGP table, a 64-bits per-VRF RD is prepended to the IP prefixes. The RD helps keep routes separate for VPNs so that customers in different VPNs do not see each others routes. It is a good practice to use the same RD for a VPN on all PEs.
- Route target (RT)—For each route, MP-BGP carries an extended community attribute (RT) that determines who is allowed to import or export that router. When a PE builds a VRF, it imports only the BGP routes that have the specific RT it is configured to import. Similarly, a VRF tells MP-BGP which RT value to use to advertise its routes, known as exporting routes.

  For intranet routing, a VPN should export and import the same RT for an optimal memory use by the BGP table. For extranet or overlapping VPNs, one VPN imports the RT of another VPN and vice versa. Route maps may be used to further tune which routes to import/export. Access to a service can also be announced using a dedicated RT.
- Virtual routing forwarding instance (VRF)—PEs use a separate routing table called VRF per-VPN. A VRF is built by using information from the BGP table built by MP-iBGP updates.
- Multi Protocol BGP (MP-iBGP) as described in RFC 2547 to create Layer 3 VPNs— Conventional BGP is designed to carry routing information for the IPv4 address family. MP-iBGP includes multi-protocol extensions such as RD, RT, and VPN label information as part of the Network Layer Reachability Information (NLRI).

  MP-iBGP carries VPNv4 routes from an ingress PE and relays it to an egress PE. At the ingress PE, MP-iBGP allocates labels for VPNv4 prefixes and installs them in the LFIB table. At the egress PE, MP-iBGP installs VPN prefixes and labels in the VRF FIB table. The associated label is appended to a packet when a packet within that VPN needs to be forwarded from the ingress PE to an egress PE. Note that this is an inner label. At the ingress PE, the outer label is derived from IGP plus LDP and is used as an outer header to switch the traffic from the ingress PE to the egress PE.
- Multi-VRF— Also known as VRF-Lite, this is a lightweight segmentation solution and a label-free solution that works without LDP and MP-iBGP. To keep traffic separate for each VPN segment, a VRF is created and mapped to a pair of ingress/egress Layer 3 interfaces. Routing information for each VPN is kept in its associated VRF instance.

Multipath Load Balancing

Current networks more commonly have redundant links and devices to the same destination. It is essential that traffic use multiple available paths for better traffic load balancing. Several load balancing mechanisms are available, including Cisco IOS software-based mechanisms such as Cisco Express Forwarding, unequal cost load balancing, OER, GLBP, eBGP, iBGP, and PBR. Per flow or per packet

Cisco Express Forwarding is employed to use multiple links on a router. Per destination (per flow) is the recommended method because per packet can introduce packet reordering and non-predictive latency within a session.

IGP can easily load balance based on the path metrics. Because BGP does not have a metric, load balancing over BGP paths becomes more challenging. Typically, BGP chooses only one best path using the complicated path selection algorithm and installs this path in the routing table. In addition, unlike what happens in IGPs, next hops of BGP routes may not be directly connected. eBGP and iBGP multipath mechanisms allow the installation of multiple BGP next hops in the routing tables (VRF routing tables).

There are two types of BGP multipath mechanisms available: eBGP and iBGP. eBGP multipath is used if a network has multiple exit points to a VPN site (destination in a VPN site). If eBGP is used to connect VPN sites to an MPLS cloud, eBGP multipath can be used on a CPE facing the cloud and iBGP multipath load balancing on PEs facing VPN sites. If IGP is used to connect VPN sites to an MPLS cloud, iBGP multipath load balancing on PEs suffices. iBGP multipath is used for dual-homed PEs. If at the egress point traffic can be sent via two PEs to the VPN site, the ingress PE needs to load balance using both exit points. iBGP multipath load balancing works with both equal cost and unequal cost paths to the destination (egress) PEs.

Note that the BGP multipath mechanism does not interfere with the BGP best path selection process; it installs multiple paths, but designates one of the paths as the best path. Multiple paths are installed in both RIB and FIB tables. Unequal cost paths are used proportionally to the link bandwidth. For BGP multipaths to work, all the path selection attributes such as weight, local preference, AS path, origin code, multi-exit discriminator (MED), and IGP distance need to be identical for both paths.

Route reflectors reflect only the best path to their clients. If the route reflectors are used to get multiple paths reflected to all the PEs, it is essential to use a unique RD value for the same VPN on each ingress PE for a VRF. Note that this does not affect RT values. For a fully-meshed VRF, the same RTs can be used for both importing and exporting VPN routes.

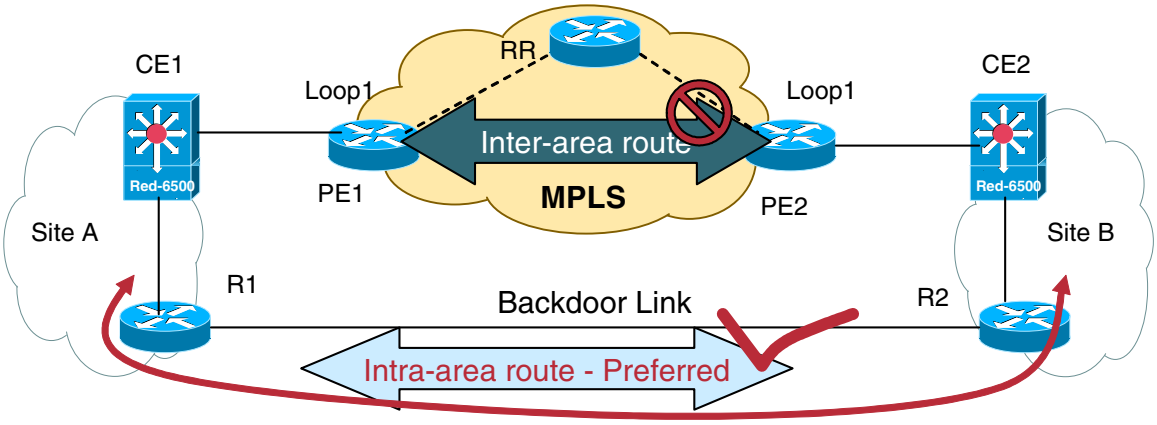
OSPF as the PE-CE Routing Protocol

When OSPF is used to connect PE and CE routers, all routing information learned from a VPN site is placed in the VPN routing and forwarding (VRF) instance associated with the incoming interface. The PE routers that attach to the VPN use BGP to distribute VPN routes to each other. When OSPF routes are propagated over the MPLS VPN backbone, additional information about the prefix in the form of BGP extended communities (route type, domain ID extended communities) is appended to the BGP update. This community information is used by the receiving PE router to decide the type of link-state advertisement (LSA) to be generated when the BGP route is redistributed to the OSPF PE-CE process. In this way, internal OSPF routes that belong to the same VPN and are advertised over the VPN backbone are seen as interarea routes on the remote sites.

OSPF and Backdoor Links

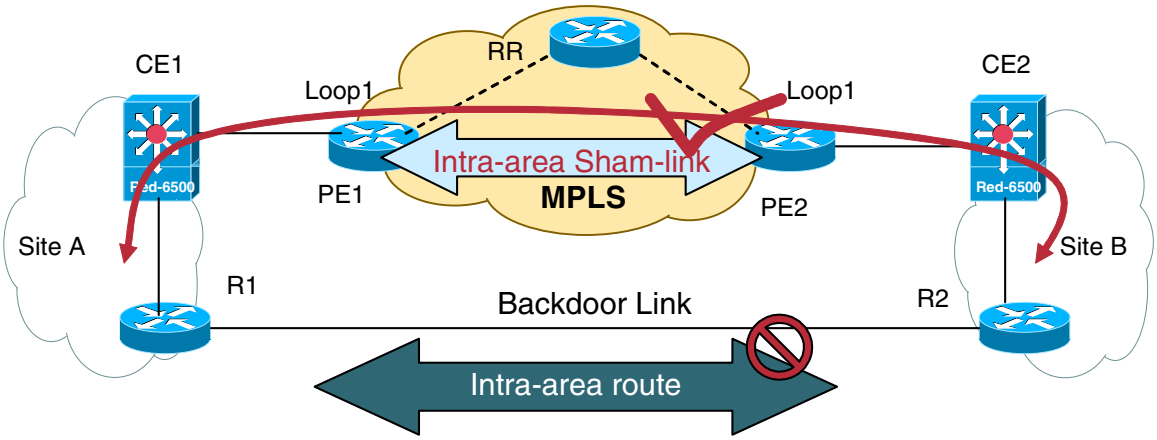
Although OSPF PE-CE connections assume that the only path between two sites is across the MPLS VPN backbone, backdoor paths between VPN sites (shown in Figure 2-3) may exist. If these sites belong to the same OSPF area, the path over a backdoor link is always selected because OSPF prefers intra-area paths to inter-area paths. (PE routers advertise OSPF routes learned over the VPN backbone as inter-area paths.) For this reason, OSPF backdoor links between VPN sites must be taken into account so that routing is performed based on policy.

Figure 2-3 OSPF—Backdoor Link Without Sham Link Support



If the backdoor links between sites are used only for backup purposes and do not participate in the VPN service, then the default route selection is not acceptable. To reestablish the desired path selection over the MPLS VPN backbone, create an additional OSPF intra-area (logical) link between ingress and egress VRFs on the relevant PE routers. This link is called a sham-link. A cost is configured with each sham-link and is used to decide whether traffic is sent over the backdoor path or the sham-link path. When a sham-link is configured between PE routers, the PEs can populate the VRF routing table with the OSPF routes learned over the sham-link.

Figure 2-4 OSPF—Backdoor Link With Sham Link Support



Because the sham-link is seen as an intra-area link between PE routers, an OSPF adjacency is created and database exchange (for the particular OSPF process) occurs across the link. The PE router can then flood LSAs between sites from across the MPLS VPN backbone. As a result, the desired intra-area connectivity is created.

Before you create a sham-link between PE routers in an MPLS VPN, you must:

1. Configure a separate /32 address on the remote PE so that OSPF packets can be sent over the VPN backbone to the remote end of the sham-link. You can use the /32 address for other sham-links.The /32 address must meet the following criteria:
   - Belong to a VRF
   - Not be advertised by OSPF

  - Be advertised by BGP

2. Associate the sham-link with an existing OSPF area.

EIGRP as PE-CE Routing Protocol

When EIGRP is used as the PE-CE protocol, EIGRP metrics are preserved across the MPLS VPN backbone though use of MP-BGP extended community attributes. The EIGRP route type and vector metric information is encoded in a series of well-known attributes. These attributes are transported across the MPLS VPN backbone and used to recreate the EIGRP route when received by the target PE router. There are no EIGRP adjacencies, EIGRP updates, or EIGRP queries sent across the MPLS VPN backbone. Only EIGRP metric information is carried across the MPLS VPN backbone via the MP-BGP extended communities.

Routes are recreated by the PE router and sent to the CE router as an EIGRP route. The same route type and cost basis as the original route are used to recreate the EIGRP route. The metric of the recreated route is increased by the link cost of the interface. On the PE router, if a route is received via BGP and the route has no extended community information for EIGRP, the route is advertised to the customer edge router as an external EIGRP route using the default metric. If no default metric is configured, the route is not advertised to the customer edge router.

EIGRP and Backdoor Links

The SoO extended community is a BGP extended community attribute that is used to identify routes that have originated from a site so that the re-advertisement of that prefix back to the source site can be prevented. The SoO extended community uniquely identifies the site from which a PE router has learned a route. SoO support provides the capability to filter MPLS VPN traffic on a per-EIGRP site basis.

If all of the routers in the customer's sites between the provider edge routers and the backdoor routers support the SoO feature, and the SoO values are defined on both the provider edge routers and the backdoor links, the provider edge routers and the backdoor routers all play a role in supporting convergence across the two (or more) sites. Routers that are not provider edge routers or backdoor routers must only propagate the SoO value on routes as they forward them to their neighbors, but they play no other role in convergence beyond the normal dual-attachment stations. The next two sections describe the operation of the PE routers and backdoor routers in this environment.

PE Router Operations

When this SoO is enabled, the EIGRP routing process on the PE router checks each received route for the SoO extended community and filters based on the following conditions:

- A received route from BGP or a CE router contains a SoO value that matches the SoO value on the receiving interface—If a route is received with an associated SoO value that matches the SoO value that is configured on the receiving interface, the route is filtered out because it was learned from another PE router or from a back door link. This behavior is designed to prevent routing loops.
- A received route from a CE router is configured with a SoO value that does not match—If a route is received with an associated SoO value that does not match the SoO value that is configured on the receiving interface, the route is accepted into the EIGRP topology table so that it can be redistributed into BGP. If the route is already installed to the EIGRP topology table but is associated with a different SoO value, the SoO value from the topology table is used when the route is redistributed into BGP.

- A received route from a CE router does not contain a SoO value—If a route is received without a SoO value, the route is accepted into the EIGRP topology table, and the SoO value from the interface that is used to reach the next hop CE router is appended to the route before it is redistributed into BGP.

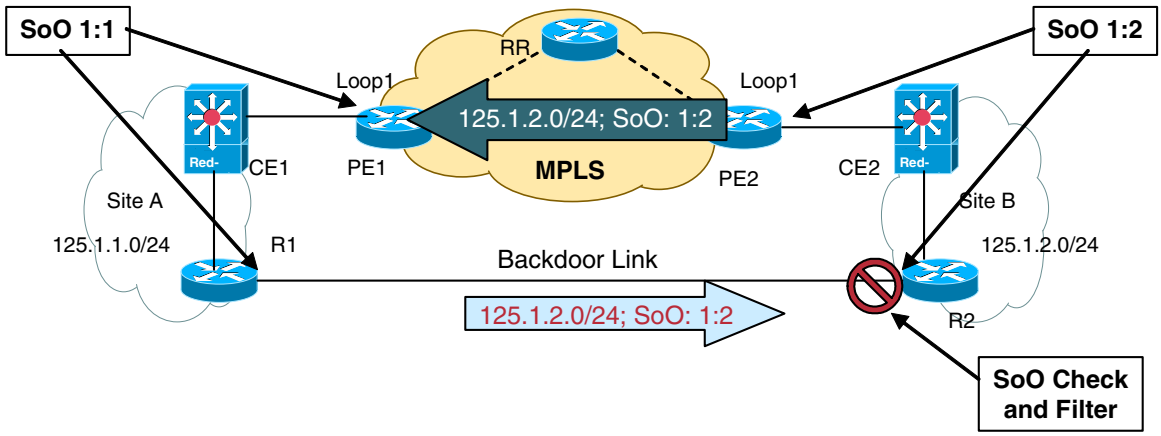
When BGP and EIGRP peers that support the SoO extended community receive these routes, they also receive the associated SoO values and pass them to other BGP and EIGRP peers that support the SoO extended community.This filtering is designed to prevent transient routes from being relearned from the originating site, which prevents transient routing loops from occurring.

The "pre-bestpath" point of insertion (POI) was introduced in the BGP Cost Community feature to support mixed EIGRP VPN network topologies that contain VPN and backdoor links. This POI is applied automatically to EIGRP routes that are redistributed into BGP. The "pre-best path" POI carries the EIGRP route type and metric. This POI influences the best path calculation process by influencing BGP to consider this POI before any other comparison step. When BGP has a prefix in the BGP table that is locally sourced and it receives the same prefix from a BGP peer, BGP compares the cost community values of the two paths. The path that has the best (or lowest) cost community value is selected as the best path.

Backdoor Link Router Operation

When a backdoor router receives EIGRP updates (or replies) from a neighbor across the backdoor link, it checks each received route to verify that it does not contain an SoO value that matches the one defined on the interface. If it finds a route with an SoO value that matches, the route is rejected and not put into the topology table. Typically, the reason that a route would be received with a matching SoO value would be that it was learned by the other site via the VPN connection and advertised back to the original site over the backdoor link. By filtering these routes based on the SoO value at the backdoor link, short term invalid routing is avoided.

Figure 2-5 EIGRP—Backdoor Link Support



In Figure 2-5, routes originating in site B are tagged with the SoO value 1:3 when the PE2 redistributes them into BGP. When the routes are redistributed from BGP into EIGRP on PE1, the SoO value is pulled out of the BGP table and retained on the routes as they are sent to site A. Routes are forwarded within site A and eventually advertised out backdoor router R1 to R2. The routes with the SoO value 1:1 are filtered out when updates are received by R2, stopping them from being relearned in Site A via the backdoor, thus preventing routing loops.

MPLS Network Convergence

Convergence can be defined as the time taken for routing nodes within a particular domain to learn about the complete topology and to recompute an alternative path (if one exists) to a particular destination after a network change has occurred. This process involves the routers adapting to these changes through synchronization of their view of the network with other routers within the same domain.

In an MPLS network, the convergence times of the following three network components can have an effect on application performance:

- Backbone convergence time—Convergence behavior in the backbone varies based on the core IGP and LDP operational mode. Core convergence time is dictated by its IGP convergence time. LDP convergence time is almost insignificant.
- VPN site convergence— Convergence behavior in a VPN site varies based on the IGP in use.
- VPN site route distribution convergence time—The redistribution delay comes from redistributing VPN site routes to MP-iBGP and redistributing MP-iBGP routes back to VPN sites. The delay is dictated by MP-iBGP.

Convergence behavior in the backbone varies based on the core IGP and LDP operational mode. Core convergence time is dictated by its IGP convergence time. LDP convergence time is almost insignificant. Convergence behavior in a VPN site varies based on the IGP in use.

The redistribution delay comes from redistributing VPN site routes to MP-iBGP and redistributing MP-iBGP routes back to VPN sites. The delay is dictated by MP-iBGP.

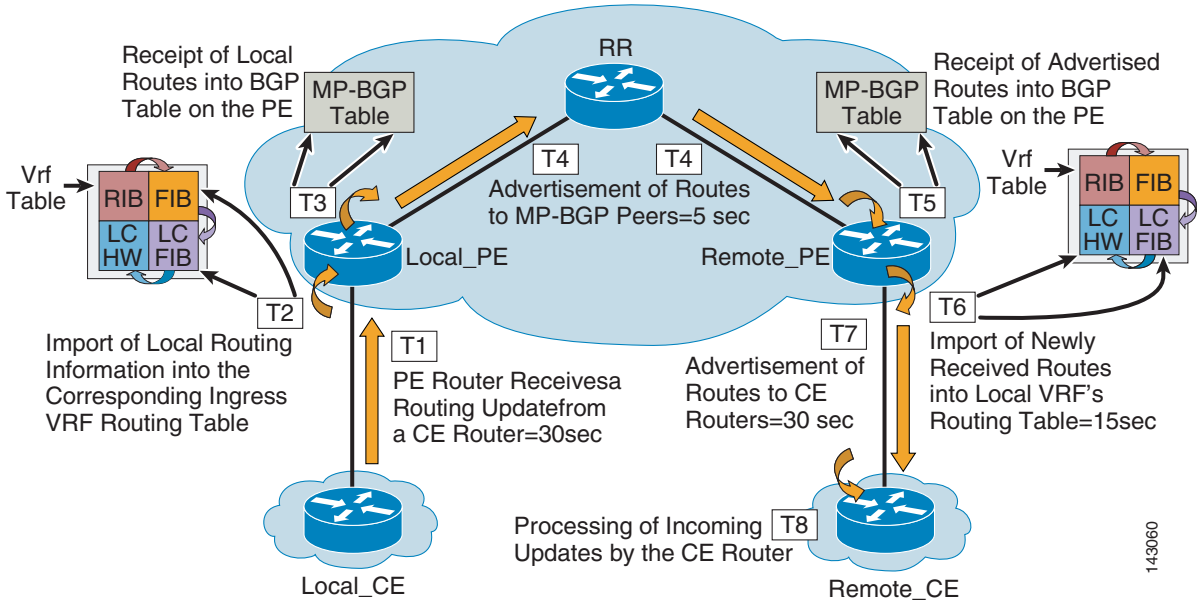
Also note that the convergence times vary for initial site updates (up convergence) and convergence occurring because of the failures in the network after the initial setup (down convergence). Only the key parameters that can be used to tune the network are highlighted in this guide.

For the most part, intelligent network design can help create faster converging networks. Some of the variables that can help tune the times are listed for each routing protocol, although the timers should be tuned after a careful examination of the current (default) network convergence times based on network design, network load, and application requirements.

Site-to-Site VPN Convergence Description with Default Timers

Figure 2-6 shows an example of site-to-site VPN convergence.

Figure 2-6 Site-to-Site VPN Convergence



The timers can be summarized in two categories:

- The first set of timers includes T1, T4, T6, and T7, which contribute higher convergence times unless tuned down.
- The second set of timers includes T2, T3, T5, and T8, and add smaller times.

Table 2-1 summarizes the maximum site-to-site convergence times with default timers for different routing protocols.

Table 2-1 Maximum Site-to-Site Convergence Times

PE-CE Protocol	Max Convergence Time (Default Settings) Where x= T2+T3+T5+T8	Max Convergence Time (Timers Tweaked Scan=5, Adv=0) Where x= T2+T3+T5+T8
BGP	~85+x seconds	~5+x seconds
OSPF	~25+x seconds	~5+x seconds
EIGRP	~25+x seconds	~5+x seconds
RIP	~85+x seconds	~5+x seconds

MPLS Network Convergence Tuning Parameters

EIGRP

EIGRP inherently provides sub-second convergence if the network is designed properly. An EIGRP network can be designed using feasible successors, summarization to bound queries, and (if applicable) using stub routers to fine tune overall network convergence time. For more information, see the following URLs:

- http://www.cisco.com/en/US/partner/tech/tk365/technologies_white_paper09186a0080094cb7.shtml

- http://www.cisco.com/application/pdf/en/us/guest/tech/tk207/c1550/cdccont_0900aecd801e4ab6.pdf
- http://www.cisco.com/en/US/partner/tech/tk365/technologies_white_paper0900aecd8023df6f.shtml

OSPF

Fast IGP convergence enhancements permit detection of link/node failure, propagation of the route change in the network, and recalculation and installation of the new routes in the routing and forwarding table as soon as possible. Some of the OSPF enhancements are OSPF event propagation, OSPF sub-second hellos tuning, OSPF LSA generation exponential backoff, and OSPF exponential backoff. LSA generation and SPF run wait times can be changed to allow faster convergence times. For more information, see the following URLs:

- http://www.cisco.com/en/US/partner/products/sw/iosswrel/ps1829/products_feature_guide09186a00 80161064.html
- http://www.cisco.com/en/US/partner/products/sw/iosswrel/ps1838/products_feature_guide09186a00 80134ad8.html

BGP

- BGP scanner time—By default, BGP scans the BGP table and routing table every 60 seconds for all the address-families that are configured under the BGP process. The next-hop validation is performed via this process. So if there is any route whose next-hop is not reachable anymore, this scan process marks the route as invalid and withdraws it. The scanner interval can be modified using the following command under vpnv4 address-family:

  ```
  bgp scan-time <5-60 seconds>
  ```

- BGP scan-time import—A number of vpnv4 routes might be learned from across the backbone, which are then subjected to best path calculation. Once best path is calculated, it gets imported into the respective VRF routing table. This import cycle runs every 15 seconds by default. Hence it can take up to a max of 15 seconds for vpnv4 routes learned by a PE from RR or another PE to make it into the local VRF routing table.

  This import process is a separate invocation and does not occur at the same time as the scan process. BGP import scan-time can be modified under vpnv4 address-family using the following command:

  ```
  bgp scan-time import <5-60 seconds>
  ```

- BGP VRF maximum-paths import—By limiting numbers of routes in a VRF, convergence time can be improved. For more information, see the following URL: http://cco.cisco.com/en/US/products/sw/iosswrel/ps5187/products_command_reference_chapter09 186a008017d029.html#wp1058523.

LDP

LDP convergence mainly depends on IGP convergence. It is insignificant compared to IGP convergence.

Bidirectional Forwarding Detection (BFD)

Bi-directional Forwarding Detection (BFD) provides rapid failure detection times between forwarding engines, while maintaining low overhead. It also provides a single, standardized method of link/device/protocol failure detection at any protocol layer and over any media. The Internet draft for

BFD defines two modes for session initiation, Active and Passive. An Active node sends BFD control packets in an effort to establish a BFD session. A Passive node does not send any BFD packets until it receives BFD packets from an Active node.

Once the BFD session and appropriate timers have been negotiated, the BFD peers send BFD control packets to each other at the negotiated interval. As long as each BFD peer receives a BFD control packet within the detect-timer period, the BFD session remains up and any routing protocol associated with BFD maintains its adjacencies. If a BFD peer does not receive a control packet within the detect interval [(Required Minimum RX Interval) * (Detect Multiplier)], it informs any clients of that BFD session (i.e., any routing protocols) about the failure. A BFD session is associated with one client only even if it is configured between the same set of peers.

BFD is currently supported for BGP, OSPF, ISIS and EIGRP protocols. Refer to the following URL for supported platforms/releases:

http://www.cisco.com/en/US/tech/tk365/technologies_white_paper0900aecd80244005.shtml

Scalability of an MPLS Network

RFC 2547 architecture calls for supporting over one million VPNs in an MPLS network, although the number of VPNs supported per PE is limited by platform resources, the type and number of services supported on the platform, CE-PE routing protocol used, traffic patterns, and traffic load. The number of CPE, PE, and P devices needed in the network depends on the size of the organization and how the sites are dispersed geographically. CPE, PE, and P devices should be sized carefully based on the network size, number of VPN sites, and traffic load. For example, memory utilized by different components:

- Hardware IDB requires 4692 Bytes (One Per Physical Interface)
- Software IDB requires 2576 Bytes (One Per Interface and Per Sub-Interface)
- MPLS Forwarding Memory (LFIB) consumes one “taginfo” (64 Bytes) per route, plus one Forwarding Entry (104 Bytes) for each path
- Minimum OSPF protocol memory needed is 168KB per process
- Need about 60-70KB per VRF and about 800-900 bytes per route
- Each BGP prefix entry with multiple iBGP paths needs 350 bytes of additional memory

MPLS Layer 2 VPNs—AToM

Any Transport over MPLS (AToM) is an industry solution for transporting Layer 2 packets over an IP/MPLS backbone. AToM is provided as part of the Unified VPN portfolio of leading-edge VPN technologies available over the widest breadth of Cisco routers and is based on the Martini draft described in the following URL:
http://www.ietf.org/Internet-drafts/draft-martini-l2circuit-trans-mpls-07.txt.

AToM is an attractive solution for the customers with ATM, Frame Relay, PPP, HDLC, or Ethernet networks that need point-to-point Layer 2 connectivity. With point-to-point virtual circuits, the Layer 2 connections retain their character as VPNs. The VPN site controls traffic routing within the network and the routing information resides on the VPN site edge router. As a result, the complexity of redistributing VPN site routing to and from the MPLS network is reduced. The MPLS PE supplies point-to-point connections or an emulated pseudowire (PW). A pseudowire is a connection between two PE devices

that connect two PW services of the same or disparate transport types. Note that Layer 2 and Layer 3 VPNs can be supported on the same PE device, but the CE-PE on a PE interface can only be a Layer 3 or Layer 2 VPN interface.

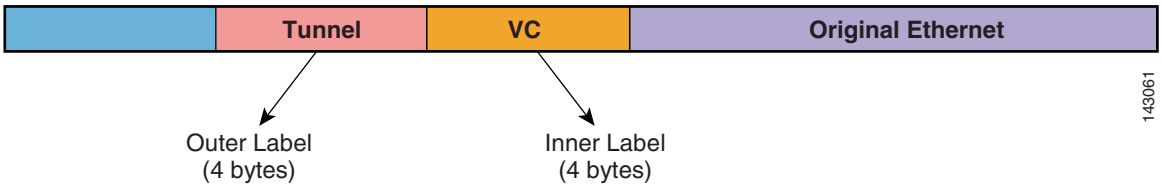
Ethernet over MPLS

Ethernet over MPLS (EoMPLS) is a popular method for creating Ethernet Virtual LAN services because it allows multiple services such as transparent LAN services (TLS) or bridging between sites, IP VPN services, and transport of desktop protocols such as SNA without interfering with the routing of the site. Ethernet traffic (unicast, broadcast, and multicast) from a source 802.1Q VLAN to a destination 802.1Q VLAN is transported over an MPLS core by mapping the VLANs to MPLS LSPs.

EoMPLS virtual circuits are created using LDP. EoMPLS uses targeted LDP sessions to dynamically set up and tear down LSPs over an MPLS core for dynamic service provisioning. No MAC address learning is required because this is a point-to-point connection that appears to be on the same wire.

Figure 2-7 shows an example of EoMPLS.

Figure 2-7 EoMPLS Example



EoMPLS comprises the following:

- Two levels of labels (8 bytes) are used:
  - Tunnel label—Outer label to forward the packet across the network
  - Virtual circuit (VC)—Inner label to bind Layer 2 interface where packets must be forwarded. The label is provided from the disposition PE. The imposition PE prepends this label so that the disposition router knows to which output interface and VC to route a packet. The egress PE uses the VC label to identify the VC and output interface to which the packet belongs.

    A VC is a 32-bit identifier used uniquely to identify the VC per tunnel and is configured between two different interfaces on PEs. The VC is used to define a point-to-point circuit over which Layer 2 PDUs are transported.

EoMPLS can operate in two modes:

- Port mode
- VLAN mode

VC type-0x0004 is used for VLAN over MPLS application and VC type-0x0005 is used for Ethernet port tunneling application (port transparency).

Port mode allows a frame coming into an interface to be packed into an MPLS packet and transported over the MPLS backbone to an egress interface. The entire Ethernet frame is transported without the preamble or FCS as a single packet.

VLAN mode transports Ethernet traffic from a source 802.1q to destination 802.1q VLAN over an MPLS core. The AToM control word is supported. However, if the peer PE does not support a control word, the control word is disabled. This negotiation is done by LDP label binding. Ethernet packets with hardware level cyclic redundancy check (CRC) errors, framing errors, and runt packets are discarded on input.

Port mode and Ethernet VLAN mode are mutually exclusive. If you enable a main interface for port-to-port transport, you cannot also enter commands on a subinterface.

EoMPLS operation is as follows:

1. The ingress PE router receives an Ethernet frame and encapsulates the packet by removing the preamble, the start of frame delimiter (SFD), and the frame check sequence (FCS). The rest of the packet header is not changed.
2. The ingress PE router adds a point-to-point virtual connection (VC) label and a label switched path (LSP) tunnel label for normal MPLS routing through the MPLS backbone.
3. The network core routers use the LSP tunnel label to move the packet through the MPLS backbone and do not distinguish Ethernet traffic from any other types of packets in the MPLS backbone.
4. At the other end of the MPLS backbone, the egress PE router receives the packet and de-encapsulates the packet by removing the LSP tunnel label if one is present. The PE router also removes the VC label from the packet.
5. The PE router updates the header, if necessary, and sends the packet out the appropriate interface to the destination switch.

QoS in AToM

The same QoS classification and marking mechanisms that are inherent in an MPLS network are used in AToM. Experimental bits in the MPLS header are used to create priority levels. For example, based on the type of service of the attachment VC, the MPLS EXP field can be set to a higher priority that allows better delivery of Layer 2 frames across the MPLS network. Layer 2 QoS, such as the 802.1P field in the IP header, can be easily mapped to MPLS EXP to translate QoS from Layer 2 to MPLS, thereby providing bandwidth, delay, and jitter guarantees. In the case of Frame Relay and ATM, the EXP values can be set by reference to the discard eligible (DE) bit marking in the frame header and to the cell loss priority (CLP) bit marking in the ATM cell header.

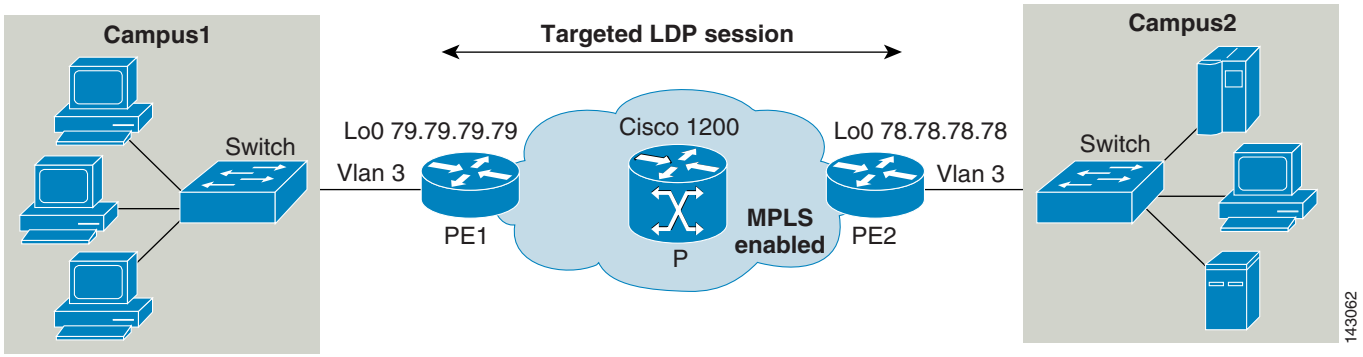
Scalability

MPLS AToM scales well because it does not require VC state information to be maintained by core MPLS devices. This is accomplished by label stacking to direct multiple connections bound for the same destination onto a single VC. The number of virtual circuits/VPNs serviced does not affect the MPLS core network. AToM, as per the IETF draft *Transport of Layer 2 Frames over MPLS,* calls for unlimited virtual circuits to be created: “This technique allows an unbounded number of Layer 2 ‘VCs’ to be carried together in a single tunnel.” Thus, it scales quite well in the network backbone. Although there are no hardware IDB limitations, the number of Layer 2 VCs supported per device (PE) is limited by the device (PE) resources, traffic load, and additional services enabled on the device (PE). From the provisioning perspective, if a fully-meshed connectivity between the sites is required, depending on the total number of sites, this solution can be labor-intensive to provision because it requires manually setting up n*(n-1) site meshes.

EoMPLS Sample Configuration

Figure 2-8 shows an example of an EoMPLS configuration topology.

Figure 2-8 EoMPLS Sample Configuration Typology



The following is the configuration on PE1:

```
mpls label protocol tdp
mpls ldp discovery directed-hello accept from 1

mpls ldp router-id Loopback0
!
interface FastEthernet2/11.3
 encapsulation dot1Q 3
 no ip directed-broadcast
 mpls l2transport route 78.78.78.78 300
 no cdp enable
!
access-list 1 permit 78.78.78.78
```

The following is the configuration on PE2:

```
mpls label protocol tdp
mpls ldp discovery directed-hello accept from 1

mpls ldp router-id Loopback0
!
interface FastEthernet2/11.3
 encapsulation dot1Q 3
 no ip directed-broadcast
 mpls l2transport route 79.79.79.79 300
 no cdp enable
!
access-list 1 permit 79.79.79.79
```

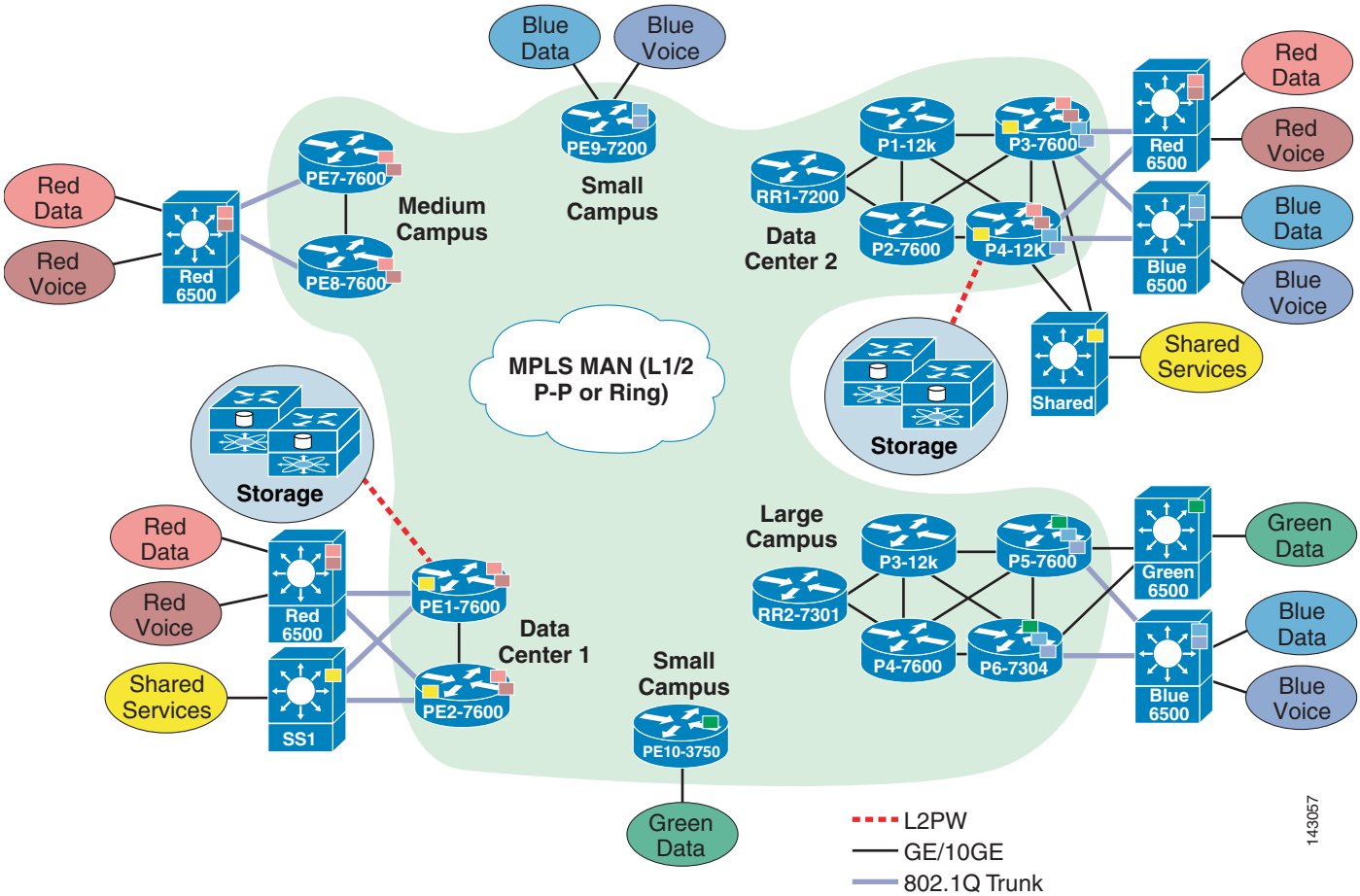
CHAPTER 3

MPLS-Based VPN MAN Reference Topology

MAN Topology

The MAN topology described in Figure 3-1 shows a network that serves three different organizations in a single corporation.

Figure 3-1 Typical Self-Managed MPLS MAN Deployment



This corporate network spans five different locations: large campus (LC), medium campus (MC), two small campuses (SC1, SC2), and two data centers (DC1, DC2).

Each organization requires that their traffic be kept separate from the traffic of any other organization. Within each organization, voice and data traffic should also be separated. To meet these requirements, traffic is segmented into five different VPNs:

- Red-data to serve data traffic in the Red organization
- Red-voice to serve voice traffic in the Red organization
- Blue-data to serve data traffic in the Blue organization
- Blue-voice to serve voice traffic in the Blue organization
- Green-data to serve the Green organization

The Blue organization resides in a large campus (LC) and a small campus 1 (SC1). The Red organization resides in a medium campus (MC). Data center 2 (DC2) serves both the Blue and Red organizations. Data center 1(DC1) serves only the Red organization. The Blue and Green organizations are using EIGRP and the Red organization is using OSPF to relay VPN subnet information.

Because voice and data traffic for each VPN user need to be segmented end-to-end, it is essential that the segmentation starts on the first Layer 3 device in the path. Note that segmentation from the access layer to a distribution layer is maintained using VLANs. Multi-VRF is deployed on the distribution layer devices (DLs), which are the first Layer 3 devices to maintain and carry the segmentation throughout the network. Users in each VLAN are directly mapped to associated VRFs on the Multi-VRF devices (DL switches) and the traffic is kept separate as it traverses through each campus and data center.

Because EIGRP and OSPF are widely adapted routing protocols in enterprise networks, testing is done with EIGRP as a core IGP and then repeated using OSPF. Note that this does not affect the VPN site routing protocols in use. The Blue and Green organizations continue to use EIGRP and the Red organization continues to use OSPF for distributing each VLAN subnet into MPLS PE regardless of the IGP used in the core.

Frame-mode unsolicited downstream on demand LDP mode is used to allocate and distribute MPLS labels. The traffic in the core is switched based on the pre-calculated incoming and outgoing labels.

Notice that dual-homing at the MPLS edge as well as redundant links in the core and distribution layer to MPLS edge exist. For example, DL1, which connects red-data and red-voice VPNs, is dual-homed to both PE1 and PE2. The distribution layer to the MPLS edge, as well as the load balancing in the core, is done by IGP Cisco Express Forwarding. To make sure the traffic is load balanced between ingress and egress edge points, MP-iBGP multipath load balancing is enabled on all the PEs. To ensure multiple paths are stored in the forwarding tables, unique RDs are used on each PE for the same VPN.

The network also has backdoor links between sites DC2 and MC for the Red organization and between sites LC and DC2 for the Blue organization. The backdoor links are not to be used for load balancing with the MPLS VPN network, but are backup links to the MPLS VPN network. To prevent routing loops for the Red organization, OSPF sham link is configured on PE3, PE4, PE7, and PE8. To prevent routing loops for the Blue organization, EIGRP SoO is configured on all the Blue site-connected PEs.

- Core AS is 1 when EIGRP is used as the IGP.
- Core area is 0 when OSPF is used as the IGP.
- MP-iBGP peers are in AS 1.
- EIGRP edge is using AS 10 for blue-data, 11 for blue-voice, and 12 for green-data.
- The Red organization uses OSPF Area 0.

VPN Information

Table 3-1 shows VPN information.

Table 3-1 VPN Information

Organization	VPN Name	RD	RT
Blue	blue-data	10:1055	10:105
	blue-voice	10:1066	10:106
Green	green-data	10:107	10:107
Red	red-data	10:1031	10:103
	red-voice	10:1042	10:104

Inventory of Devices

Table 3-2 lists an inventory of devices.

Table 3-2 Inventory of Devices

	Reference	Actual	Loop0
DC1	PE1	7600-DC1-PE1	125.1.125.5/32
	PE2	7600-DC1-PE2	125.1.125.6/32
	SS1	7600-DC1-SS1	125.1.125.17/32
DC2	P1	12k-DC2-P1	125.1.125.1/32
	P2	7600-DC2-P2	125.1.125.2/32
	RR1	7200-DC2-RR1	125.1.125.15/32
	PE4	12k-DC2-PE4	125.1.125.8/32
	PE3	7600-DC2-PE3	125.1.125.7/32
LC	P3	12k-LC-P3	125.1.125.3/32
	PE5	7600-LC-PE5	125.1.125.9/32
	PE6	7304-LC-PE6	125.1.125.10/32
	RR2	7200-LC-RR2	125.1.125.16/32
	P4	7600-LC-P4	125.1.125.4/32
	SS2	7600-LC-SS2	125.1.125.18/32
MC	PE8	7600-MC-PE8	125.1.125.12/32
	PE7	7600-MC-PE7	125.1.125.11/32
SC1	PE9	7200-SC1-PE9	125.1.125.13/32
SC2	PE10	3750-SC2-PE10	125.1.125.14/32

Building a MAN MPLS VPN Network

Layer 1 and some Layer 2 as well as IP addresses are enabled on the interfaces. Interfaces are in up and up spoofing mode.

To build a MAN MPLS VPN network, complete the following steps:

Step 1 Build the MPLS core:

a. Enable EIGRP (or OSPF) on the core routers, RR, and PE core-facing interfaces.

b. Enable Cisco Express Forwarding:

```
Router(config)# ip cef
```

c. Select the LDP router id. Enable LDP on the core routers, RR, and PE core-facing interfaces:

```
Router(config)#mpls ldp router-id loopback0 force
Router(config)#interface interface #
Router(config)#mpls ip
Router(config)#mpls label protocol ldp
```

Step 2 Build the MPLS Layer 3 VPNs:

a. Enable MP-iBGP on PEs and establish BGP peering among all the PE routers and RRs.

Although route reflectors are not necessary, they are used for scalability purpose. BGPv4 requires that all the iBGP devices be fully meshed. With route reflector in use, PEs would only have to peer with a route reflector instead of peering with each other. This reduces having to fully mesh PEs by n(n-1), n being total numbers of PEs. As the RRs do not need to be in the data path, they could be local on any router that does not require a powerful switching capacity.

When route reflectors are in use, and all the outgoing updates have the same policy, it further helps fine tune the network by using peer-groups on route reflectors. This reduces the number of outgoing updates (per client) that a route reflector has to generate. Here two route reflectors are used for high availability purpose. All the PEs peer with RR instead of with each other.

PEs peer with route reflectors to exchange VPNv4 routing information:

```
7200-DC2-RR1(config)#router bgp 1
7200-DC2-RR1(config-router)#
```

Route reflectors peer with PEs to reflect VPNv4 routing information learned from other PEs:

```
Peer-Group Setup:
7200-DC2-RR1(config)#router bgp 1
7200-DC2-RR1(config-router)#neighbor CampusPE peer-group
       7200-DC2-RR1(config)#neighbor CampusPE remote-as 1
7200-DC2-RR1(config)#neighbor CampusPE update-source
Loopback0

7200-DC2-RR1(config)#neighbor <PE loopback#> peer-group CampusPE
```

VPNv4 BGP peering between PEs and RRs on a route reflector:

```
7200-DC2-RR1(config-router)#  address-family vpnv4
7200-DC2-RR1(config-router-af)# neighbor CampusPE activate
7200-DC2-RR1(config-router-af)# neighbor CampusPE route-reflector-client

7200-DC2-RR1(config-router-af)# neighbor CampusPE send-community extended

7200-DC2-RR1(config-router-af)# neighbor <PE loopback#> peer-group CampusPE
```

VPNv4 BGP peering between PEs and RRs on a PE.

Enable PEs to exchange VPNv4 routing information to the RRs.

```
7600-DC1-PE1(config)#router bgp 1
7600-DC1-PE1(config-router)#no synchronization
7600-DC1-PE1(config-router)#bgp log-neighbor-changes
7600-DC1-PE1(config-router)#neighbor <RR1 loopback ip#> remote-as 1

7600-DC1-PE1(config-router)#neighbor <RR1 loopback ip#> update-source Loopback0

7600-DC1-PE1(config-router)# address-family vpnv4
7600-DC1-PE1(config-router-af)# neighbor <RR1 loopback ip#> activate

7600-DC1-PE1(config-router-af)# neighbor 125.1.125.15 send-community extended
```

If the network does not have RRs, set up VPNv4 peering with other PEs in the network by using the PEs loopback IP addresses. It is important to set up BGP peering before VPN site routing information is redistributed into BGP for easier management and troubleshooting of the network.

b. Create a VPN.

MPLS allows support for multiple VPNs in a scalable way. VLANs terminating on the distribution layer can be individually mapped into a VRF with its own routing instance. The VRF-name is a unique and case-sensitive value. It is used to identify the VRF.

```
7600-DC1-PE1(config)#ip vrf vrf name
7600-DC1-PE1(config)#ip vrf red-data
```

c. Create an RD under its associated VRF.

Use a unique RD per VRF.

```
7600-DC1-PE1(config-vrf)#rd route-distinguisher unique value
7600-DC1-PE1(config-vrf)#rd 10:1031
```

Note You can assign only one RD to a VRF. If an RD needs to be changed for any reason after VRFs are operational, make sure to save the entire VRF-related configuration. Changing the RD requires removing the current RD, which removes the associated VRF. VPN site routes as well as MP-iBGP redistribution also need to be reconfigured.

d. Create an RT under its associated VRF.

```
7600-DC1-PE1(config-vrf)#route-target {import | export | both} route- target
7600-DC1-PE1(config-vrf)#route-target export 10:103
7600-DC1-PE1(config-vrf)#route-target import 10:103
```

e. Bind a VRF to an interface.

To specify the interfaces belonging to a VPN (VRF), use the following command:

```
Router(config)#interface interface #
Router(config-if)#ip vrf forwarding <vrf-name>
7600-DC1-PE1(config)#interface GigabitEthernet1/7.1
7600-DC1-PE1(config-subif)#  ip vrf forwarding red-data
```

Note You can only bind one VRF to an interface. After an interface is bound to a VRF, its IP address is not be part of the global routing table. You need to examine the associated VRF instance.

f. Redistribute VPN site routing information into MP-iBGP.

Reachability information for the adjacent VPN sites at the ingress PE needs to be sent to the egress PE so that it can update its adjacent VPN sites. To establish the connectivity between two VPN sites, redistribute routing into MP-iBGP at the ingress PE and back to the remote site IGP at the egress PE.

For OSPF:

```
!
7600-DC1-PE1(config)#router bgp 1
7600-DC1-PE1(config-router)#address-family ipv4 vrf red-data

7600-DC1-PE1(config-router-af)# redistribute ospf 2 vrf red-data match internal
external 1 external 2

7600-DC1-PE1(config-router-af)# maximum-paths ibgp unequal-cost 6

7600-DC1-PE1(config-router-af)# no auto-summary
7600-DC1-PE1(config-router-af)# no synchronization
7600-DC1-PE1(config-router-af)# exit-address-family
!
```

For EIGRP:

```
!
7600-DC2-PE3(config)#router bgp 1
7600-DC2-PE3(config-router)#address-family ipv4 vrf blue-voice

7600-DC2-PE3(config-router-af)#redistribute eigrp 11
7600-DC2-PE3(config-router-af)#maximum-paths ibgp unequal-cost 8

7600-DC2-PE3(config-router-af)#no auto-summary
7600-DC2-PE3(config-router-af)#no synchronization
7600-DC2-PE3(config-router-af)#exit-address-family
!
```

g. Redistribute remote site routing information learned via MP-iBGP into local site IGP.

Configuration varies based on the routing protocol used in VPN sites.

For OSPF:

```
!
7600-DC1-PE1(config)#router ospf 1 vrf red-data
 7600-DC1-PE1(config-router)#log-adjacency-changes
        7600-DC1-PE1(config-router)#redistribute bgp 1 subnets
7600-DC1-PE1(config-router)#network <site's subnet> area 0
!
```

Note OSPF process 1 is used for VRF red-data. The routes learned from remote VPN site via BGP are distributed into OSPF process 1 to update adjacent VPN segments. A different process number would be used to support additional VPN.

For EIGRP:

```
       !
7600-DC2-PE3(config)#router eigrp 10
7600-DC2-PE3(config-router)# address-family ipv4 vrf blue-voice

7600-DC2-PE3(config-router-af)#redistribute bgp 1 metric 1000000 100 255 1 1500

 7600-DC2-PE3(config-router-af)#network <VPN site Subnet or network #>

 7600-DC2-PE3(config-router-af)#maximum-paths 8
```

```
7600-DC2-PE3(config-router-af)#no auto-summary
7600-DC2-PE3(config-router-af)#autonomous-system 11
 7600-DC2-PE3(config-router-af)#exit-address-family
      !
```

h. For OSPF VPN sites with backdoor links, configure a sham-link on a pair of ingress/egress PEs:

- Configure an additional /32 loopback interface and bind the associated VRF to this loopback interface:

```
interface Loopback1
ip vrf forwarding red-data
ip address 125.1.125.103 255.255.255.255
```

- Advertise the loopback interface address through BGP and not through OSPF process:

```
7600-DC2-PE3(config)#router bgp 1
7600-DC2-PE3(config-router)# address-family ipv4 vrf red-data
7600-DC2-PE3(config-router-af)#redistribute connected metric 1
```

- Associate the sham-link with an existing OSPF area and configure under the associated VRF OSPF process between a pair of ingress egress PEs:

```
7600-DC2-PE3(config)#router ospf 1 vrf red-data
7600-DC2-PE3(config-router)#area 0 sham-link 125.1.125.103 125.1.125.107
7600-DC2-PE3(config-router)# area 0 sham-link 125.1.125.103 125.1.125.108
```

Note The sham-link is set up between PE3 and PE5 and PE3 and PE6 as the backdoor link exists between these two sites.

i. For EIGRP sites with backdoor links, configure SoO on PE and CE interfaces.

Step 3 Enable segmentation on multi-VRF devices.

Note This is done on a distribution layer device. Access layer VLANs are terminated and mapped into the associated VRF on a distribution layer device (DL1). End-to-end segmentation across multiple campuses is achieved by maintaining this segmentation using dedicated interfaces for each VPN subnet to connect to the MPLS ingress PE.

a. Create VRFs on the distribution layer device.

```
!
ip vrf red-data
 rd 10:103
!
ip vrf red-voice
 rd 10:104
!
```

b. Bind a VRF to a pair of ingress-egress interfaces.

Ingress interfaces connecting to VLAN:

```
!
interface GigabitEthernet5/7
 description To RT - port 103/1
 no ip address
interface GigabitEthernet5/7.1
 encapsulation dot1Q 505
 ip vrf forwarding red-data
 ip address 125.1.1.65 255.255.255.224
```

```
!
interface GigabitEthernet5/7.2
 encapsulation dot1Q 506
 ip vrf forwarding red-voice
 ip address 125.1.10.9 255.255.255.252
!
```

Egress interfaces connecting to PE:

```
!
interface GigabitEthernet5/7.1
 encapsulation dot1Q 505
 ip vrf forwarding red-data
 ip address 125.1.1.65 255.255.255.224
!
interface GigabitEthernet5/7.2

encapsulation dot1Q 506
 ip vrf forwarding red-voice
 ip address 125.1.10.9 255.255.255.252
!
```

c. Redistribute VPN site routing information.

Use separate routing processes per VPN to exchange routing information with PEs:

```
!
router ospf 1 vrf red-data
 log-adjacency-changes
 capability vrf-lite
 redistribute static subnets
 network <> area <>
 maximum-paths 6
!
router ospf 2 vrf red-voice
 log-adjacency-changes
 capability vrf-lite
 redistribute static subnets
 network <> area <>
 maximum-paths 6
!
```

Implementing Advanced Features on MPLS-Based VPNs

QoS for Critical Applications

QoS Design Overview

Next generation (NG)-WAN/MAN networks form the backbone of business-ready networks. These networks transport a multitude of applications, including real-time voice, high-quality video, and delay-sensitive data. NG-WAN/MAN networks must therefore provide predictable, measurable, and sometimes guaranteed services by managing bandwidth, delay, jitter, and loss parameters on a network via QoS technologies.

QoS technologies refer to the set of tools and features available within Cisco hardware and software to manage network resources; these include classification and marking tools, policing and shaping tools, congestion management and congestion avoidance tools, as well as link-efficiency mechanisms. QoS is considered the key enabling technology for network convergence. The objective of QoS technologies is to make voice, video, and data convergence appear transparent to end users. QoS technologies allow different types of traffic to contend inequitably for network resources. Voice, video, and critical data applications may be granted priority or preferential services from network devices so that the quality of these strategic applications does not degrade to the point of being unusable. Therefore, QoS is a critical, intrinsic element for successful network convergence. However, QoS tools are not only useful in protecting desirable traffic, but also in providing deferential services to undesirable traffic such as the exponential propagation of worms.

A successful QoS deployment is comprised of multiple phases, including the following:

- Strategically defining the business objectives to be achieved via QoS.
- Analyzing the service level requirements of the various traffic classes for which to be provisioned.
- Designing and testing QoS policies before production network rollout.
- Rolling out the tested QoS designs to the production network.
- Monitoring service levels to ensure that the QoS objectives are being met.

These phases may need to be repeated as business conditions change and evolve. The following sections focus on the first three phases of a QoS deployment and specifically adapt best-practice QoS design to the NG-WAN/MAN.

Strategically Defining the Business Objectives

QoS technologies are the enablers for business/organizational objectives. Therefore, the way to begin a QoS deployment is not to activate QoS features simply because they exist, but to start by clearly defining the objectives of the organization. For example, among the first questions that arise during a QoS deployment are the following: How many traffic classes should be provisioned for? What should they be?

To help answer these fundamental questions, organizational objectives need to be defined, such as the following:

- Is the objective to enable VoIP only or is video also required?
- If video is required, is video-conferencing required or streaming video? Or both?
- Are there applications that are considered mission-critical and if so, what are they?
- Does the organization wish to squelch certain types of traffic and if so, what are they?

To help address these crucial questions and to simplify QoS, Cisco has adopted a new initiative called the "QoS Baseline." The QoS Baseline is a strategic document designed to unify QoS within Cisco from enterprise to service provider and from engineering to marketing. The QoS Baseline was written by the most qualified Cisco QoS experts, who have developed or contributed to the related IETF RFC standards (as well as other technology standards) and are thus eminently qualified to interpret these standards. The QoS Baseline also provides uniform, standards-based recommendations to help ensure that QoS designs and deployments are unified and consistent. The QoS Baseline defines up to 11 classes of traffic that may be viewed as critical to a given enterprise. A summary these classes and their respective standards-based marking recommendations are presented in Table 4-1.

Table 4-1 Cisco QoS Baseline/Technical Marketing (Interim) Classification and Marking Recommendations

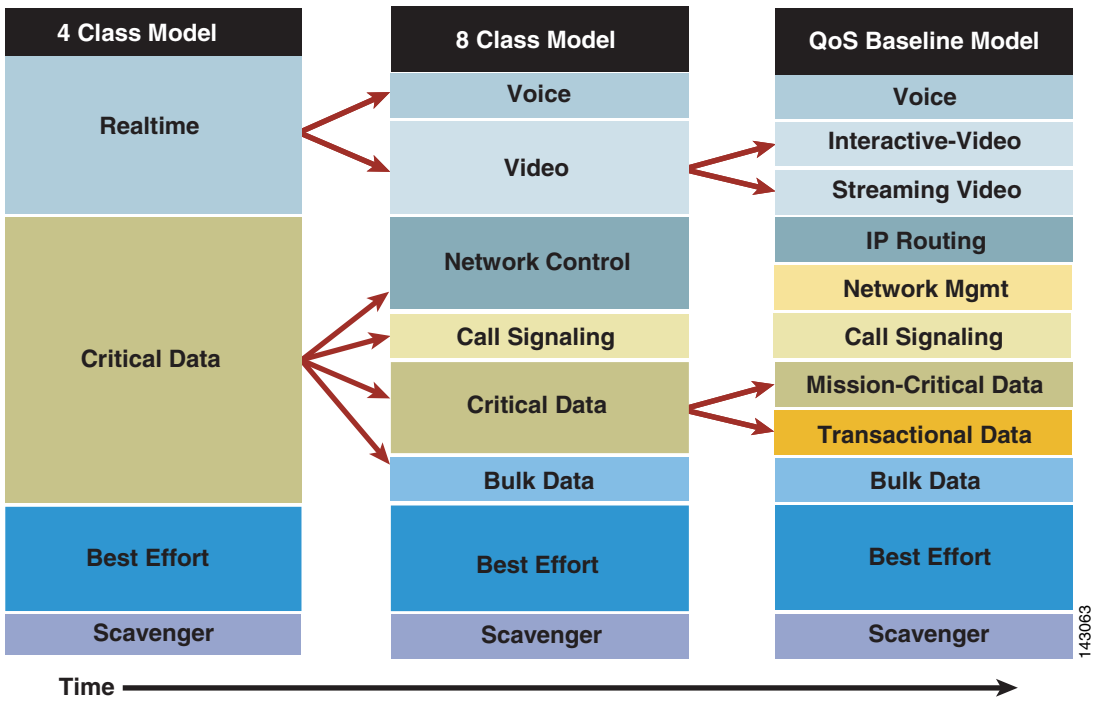
Application	Classification		Referencing Standard	Recommended Configuration
	PHB	DSCP		
IP Routing	CS6	48	RFC 2474-4.2.2	Rate-based Queuing + RED
Voice	FF	46	RFC 3246	RSVP Admission Control + Priority
Interactive-Video	AF 41	34	RFC 2957	RSVP + Rate-Based Queuing + DSCP
Streaming Video	CS4	32	RFC 2474-4.2.2	RSVP + Rate-Based Queuing + RED
Mission-Critical	AF 31	26	RFC 2597	Rate-Based Queuing + DSCP-WRED
Call Signaling	CS3	24	RFC 2474-4.2.2	Rate-Based Queuing + RED
Transactional Data	AF 21	18	RFC 2597	Rate-Based Queuing + DSCP-WRED
Network Mgmt	CS2	16	RFC 2474-4.2.2	Rate-based Queuing + RED
Bulk Data	AF 11	10	RFC 2597	Rate-Based Queuing + DSCP-WRED
Scavenger	CS1	8	Internet 2	No BW Guarantee + RED
Best Effort	0	0	RFC 2474-4.1	BW Guarantee Rate-Based Queuing

Note The QoS Baseline recommends marking Call-Signaling to CS3. However, originally Cisco IP telephony products marked Call-Signaling to AF31. A marking migration is under way within Cisco to change all IP telephony products to mark Call-Signaling to CS3 by default. For companies deploying IP telephony products that still might be using AF31 for Call-Signaling, a recommended interim marking strategy is to use both AF31 and CS3 for Call-Signaling marking and to mark Locally-Defined Mission-Critical Data applications to a temporary placeholder (non-standard) DSCP, such as 25. Upon completion of the migration, the QoS Baseline marking recommendations of CS3 for Call-Signaling and AF31 for Locally-Defined Mission-Critical Data applications should be used. These marking recommendations are more in line with RFC 2474 and RFC 2597.

Enterprises do not need to deploy all 11 classes of the QoS Baseline model. This model is intended to be a forward-looking guide that considers as many classes of traffic with unique QoS requirements as possible. Familiarity with this model can assist in the smooth expansion of QoS policies to support additional applications as future requirements arise.

However, at the time of QoS deployment, the enterprise needs to clearly define their organizational objectives, which correspondingly determine how many traffic classes are required. This consideration should be tempered with the determination of how many application classes the networking administration team feels comfortable with deploying and supporting. Platform-specific constraints or service provider constraints may also affect the number of classes of service. At this point, you should also consider a migration strategy to allow the number of classes to be smoothly expanded as future needs arise, as shown in Figure 4-1.

Figure 4-1 Example Strategy for Expanding the Number of Classes of Service over Time



Platform limitations do not necessarily have to be considered as gating factors to the number of classes that can be supported over the NG-WAN/MAN. On platforms that support more classes, these may be configured; on platforms that support fewer classes, some classes must be collapsed to accommodate hardware limitations. The main consideration is that policies need to be kept consistent and complementary to achieve expected per-hop behaviors.

A strategic standards-based guide such as the QoS Baseline coupled with a working knowledge of QoS tools and syntax is a prerequisite for any successful QoS deployment. However, you must also understand the service level requirements of the various applications requiring preferential or deferential treatment within the network.

Analyzing the Service Level Requirements

QoS Requirements of VoIP

This section includes the following topics:

- Voice (bearer traffic)
- Call -Signaling traffic

VoIP deployments require provisioning explicit priority servicing for VoIP (bearer stream) traffic and a guaranteed bandwidth service for Call-Signaling traffic. These related classes are examined separately.

VoIP (Bearer) Traffic

The following is a summary of the key QoS requirements and recommendations for Voice (bearer traffic):

- Voice traffic should be marked to DSCP EF per the QoS Baseline and RFC 3246.
- Loss should be no more than 1 percent.
- One-way latency (mouth-to-ear) should be no more than 150 ms.
- Average one-way jitter should be targeted under 30 ms.
- 21–320 kbps of guaranteed priority bandwidth is required per call (depending on the sampling rate, VoIP codec, and Layer 2 media overhead).

Voice quality is directly affected by all three QoS quality factors of loss, latency, and jitter.

Loss causes voice clipping and skips. The packetization interval determines the size of samples contained within a single packet. Assuming a 20 ms (default) packetization interval, the loss of two or more consecutive packets results in a noticeable degradation of voice quality. VoIP networks are typically designed for very close to zero percent VoIP packet loss, with the only actual packet loss being because of Layer 2 bit errors or network failures.

Excessive latency can cause voice quality degradation. The goal commonly used in designing networks to support VoIP is the target specified by ITU standard G.114, which states that 150 ms of one-way, end-to-end (mouth-to-ear) delay ensures user satisfaction for telephony applications. A design should apportion this budget to the various components of network delay (propagation delay through the backbone, scheduling delay because of congestion, and the access link serialization delay) and service delay (because of VoIP gateway codec and de-jitter buffer).

If the end-to-end voice delay becomes too long, the conversation begins to sound like two parties talking over a satellite link or even a CB radio. Although the ITU G.114 states that a 150 ms one-way (mouth-to-ear) delay budget is acceptable for high voice quality, lab testing has shown that there is a negligible difference in voice quality mean opinion scores (MOS) using networks built with 200 ms

delay budgets. Cisco thus recommends designing to the ITU standard of 150 ms, but if constraints exist where this delay target cannot be met, then the delay boundary can be extended to 200 ms without significant impact on voice quality.

Jitter buffers (also known as play-out buffers) are used to change asynchronous packet arrivals into a synchronous stream by turning variable network delays into constant delays at the destination end systems. The role of the jitter buffer is to balance the delay and the probability of interrupted play-out because of late packets. Late or out-of-order packets are discarded.

If the jitter buffer is either set arbitrarily large or arbitrarily small, then it imposes unnecessary constraints on the characteristics of the network. A jitter buffer set too large adds to the end-to-end delay, meaning that less delay budget is available for the network such that the network needs to support a delay target tighter than practically necessary. If a jitter buffer is too small to accommodate the network jitter, then buffer underflows or overflows can occur.

An underflow occurs when the buffer is empty when the codec needs to play out a sample. An overflow occurs when the jitter buffer is already full and another packet arrives that cannot therefore be queued in the jitter buffer. Both jitter buffer underflows and overflows cause packets to be discarded.

Adaptive jitter buffers aim to overcome these issues by dynamically tuning the jitter buffer size to the lowest acceptable value. Where such adaptive jitter buffers are used, you can in theory engineer out explicit considerations of jitter by accounting for worst-case per hop delays. Advanced formulas can be used to arrive at network-specific design recommendations for jitter based on maximum and minimum per-hop delays. Alternatively, a 30 ms value can be used as a jitter target because extensive lab testing has shown that when jitter consistently exceeds 30 ms, voice quality degrades significantly.

Because of its strict service level requirements, VoIP is well suited to the expedited forwarding per-hop behavior, as defined in RFC 3246 (formerly RFC 2598). It should therefore be marked to DSCP EF (46) and assigned Strict Priority servicing at each node, regardless of whether such servicing is done in hardware (as in the Cisco 7600 or 12000 routers via hardware priority queuing) or in software (as in Cisco 7200 routers via LLQ).

Call-Signaling Traffic

The following are key QoS requirements and recommendations for Call-Signaling traffic:

- Call-Signaling traffic should be marked as DSCP CS3 per the QoS Baseline (during migration, it may also be marked the legacy value of DSCP AF31).
- 150 bps (plus Layer 2 overhead) per phone of guaranteed bandwidth is required for Voice control traffic; more may be required, depending on the call signaling protocol(s) in use.

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 can result in noticeable delay-to-dial-tone (DDT) and lengthy call setup times, both of which generally translate to poor user experiences.

The 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/aggressive-dropping per-hop behaviors. Most Cisco IP telephony products have already begun transitioning to DSCP CS3 for Call-Signaling marking. If the enterprise is still in a migration between older and new IP telephony products and software, during the interim period both code points (CS3 and AF31) should be reserved for Call-Signaling marking until the transition is complete.

QoS Requirements of Video

This section describes the two main types of video traffic and includes the following topics:

- Interactive-Video
- Streaming Video

Interactive-Video

When provisioning for Interactive-Video (IP videoconferencing) traffic, the following guidelines are recommended:

- Interactive-Video traffic should be marked to DSCP AF41; excess Interactive-Video traffic can be marked down by a policer to AF42 or AF43.
- Loss should be no more than 1 percent.
- One-way latency should be no more than 150 ms.
- Jitter should be no more than 30 ms.
- Interactive-Video queues should be overprovisioned by 20 percent to accommodate bursts.

Because IP Videoconferencing (IP/VC) includes an audio codec for voice and relies on extending a real-time user experience to the video conference, it has the same loss, delay, and delay variation requirements as voice, but the traffic patterns of videoconferencing are radically different from voice.

Because (unlike VoIP) IP/VC packet sizes and rates vary given the motion-based nature of the video codec, the header overhead percentage varies as well, so an absolute value of bandwidth utilization and overhead cannot be accurately calculated for all streams. Testing, however, has shown a conservative rule of thumb for IP/VC bandwidth provisioning is to overprovision the guaranteed/priority bandwidth by 20 percent over the video call rate, which accounts for Layer 2 and Layer 3 overhead and a maximum transmission rate. For example, a user running a 384kbps video call (64kbps audio, 320 kbps video) uses a maximum bandwidth of 384kbps plus 20 percent, for approximate peak usage of 460kbps.

Streaming Video

When addressing the QoS needs of Streaming Video traffic, the following guidelines are recommended:

- Streaming Video (whether unicast or multicast) should be marked to DSCP CS4 as designated by the QoS Baseline.
- Loss should be no more than 5 percent.
- Latency should be no more than 4–5 seconds (depending on video application buffering capabilities).
- There are no significant jitter requirements.
- Guaranteed bandwidth (CBWFQ) requirements depend on the encoding format and rate of the video stream.

Streaming video applications have more lenient QoS requirements because they are delay-insensitive (the video can take several seconds to cue-up) and are largely jitter-insensitive (because of application buffering). However, streaming video may contain valuable content, such as e-learning applications or multicast company meetings, and therefore may require service guarantees.

The QoS Baseline recommendation for Streaming Video marking is DSCP CS4.

Non-organizational video content (or video that is strictly entertainment-oriented in nature such as movies, music videos, humorous commercials, and so on) might be considered for a ("less-than-Best-Effort") Scavenger service. This means that these streams play if bandwidth exists, but they are the first to be dropped during periods of congestion.

QoS Requirements of Data

There are hundreds of thousands of data networking applications. Some are TCP, others are UDP; some are delay sensitive, others are not; some are bursty in nature, others are steady; some are lightweight, others require high bandwidth, and so on. Not only do applications vary one from another, but even the same application can vary significantly between versions.

Given this, determining how to best provision QoS for data is a daunting proposition. The Cisco QoS Baseline identifies four main classes of data traffic, according to their general networking characteristics and requirements:

- Best Effort
- Bulk Data
- Transactional/Interactive Data
- Locally-Defined Mission-Critical Data

Best Effort Data

The Best Effort class is the default class for all data traffic. An application is removed from the default class only if it has been selected for preferential or deferential treatment.

When addressing the QoS needs of Best Effort data traffic, Cisco recommends the following guidelines:

- Best Effort traffic should be marked to DSCP 0.
- Adequate bandwidth should be assigned to the Best Effort class as a whole, because the majority of applications default to this class; reserve at least 25 percent for Best Effort traffic.

Typical enterprises have several hundred, if not thousands, of data applications running over their networks (the majority of which default to the Best Effort class). Therefore, you need to provision adequate bandwidth for the default class as a whole to handle the sheer volume of applications that will be included in it. Otherwise, applications defaulting to this class are easily drowned out, which typically results in an increased number of calls to the networking help desk from frustrated users. Cisco therefore recommends that you reserve at least 25 percent of link bandwidth for the default Best Effort class.

Bulk Data

The Bulk Data class is intended for applications that are relatively non-interactive and drop-insensitive and that typically span their operations over a long period of time as background occurrences. Such applications include the following:

- FTP
- E-mail
- Backup operations
- Database synchronizing or replicating operations
- Content distribution
- Any other type of background operation

When addressing the QoS needs of Bulk Data traffic, Cisco recommends the following guidelines:

- Bulk Data traffic should be marked to DSCP AF11; excess Bulk Data traffic can be marked down by a policer to AF12; violating bulk data traffic may be marked down further to AF13 (or dropped).
- Bulk Data traffic should have a moderate bandwidth guarantee, but should be constrained from dominating a link.

The advantage of provisioning moderate bandwidth guarantees to Bulk Data applications rather than applying policers to them is that Bulk Data applications can dynamically take advantage of unused bandwidth and thus speed up their operations during non-peak periods. This in turn reduces the likelihood of their bleeding into busy periods and absorbing inordinate amounts of bandwidth for their time-insensitive operations.

Transactional/Interactive Data

The Transactional/Interactive Data class, also referred to simply as Transactional Data, is a combination of two similar types of applications: Transactional Data client-server applications and Interactive Messaging applications.

The response time requirement separates Transactional Data client-server applications from generic client-server applications. For example, with Transactional Data client-server applications such as SAP, PeopleSoft, and Data Link Switching (DLSw+), the transaction is a foreground operation; the user waits for the operation to complete before proceeding.

E-mail is not considered a Transactional Data client-server application, because most e-mail operations occur in the background and users do not usually notice even several hundred millisecond delays in mailspool operations.

When addressing the QoS needs of Transactional Data traffic, Cisco recommends the following guidelines:

- Transactional Data traffic should be marked to DSCP AF21; excess Transactional Data traffic can be marked down by a policer to AF22; violating Transactional Data traffic can be marked down further to AF23 (or dropped).
- Transactional Data traffic should have an adequate bandwidth guarantee for the interactive, foreground operations they support.

Locally-Defined, Mission-Critical Data

The Locally-Defined Mission-Critical Data class is probably the most misunderstood class specified in the QoS Baseline. Under the QoS Baseline model, all traffic classes (with the exclusion of Scavenger and Best Effort) are considered critical to the enterprise. The term “locally-defined” is used to underscore the purpose of this class, which is to provide each enterprise with a premium class of service for a select subset of their Transactional Data applications that have the highest business priority for them.

For example, an enterprise may have properly provisioned Oracle, SAP, BEA, and DLSw+ within their Transactional Data class. However, the majority of their revenue may come from SAP, and therefore they may want to give this Transactional Data application an even higher level of preference by assigning it to a dedicated class such as the Locally-Defined Mission-Critical Data class.

Because the admission criteria for this class is non-technical (being determined by business relevance and organizational objectives), the decision of which applications should be assigned to this special class can easily become an organizationally- and politically-charged debate. Cisco recommends that you assign as few applications to this class from the Transactional Data class as possible. You should also obtain executive endorsement for application assignments to the Locally-Defined Mission-Critical Data class, because the potential for QoS deployment derailment exists without such an endorsement.

For the sake of simplicity, this class is referred to simply as Mission-Critical Data. When addressing the QoS needs of Mission-Critical Data traffic, Cisco recommends the following guidelines:

- Mission-Critical Data traffic should be marked to DSCP AF31; excess Mission-Critical Data traffic can then be marked down by a policer to AF22 or AF23. However, DSCP AF31 is currently being used by Cisco IP telephony equipment as Call-Signaling, so until all Cisco IPT products mark Call-Signaling to DSCP CS3, a temporary placeholder code point (DSCP 25) can be used to identify Mission-Critical Data traffic.
- Mission-Critical Data traffic should have an adequate bandwidth guarantee for the interactive, foreground operations they support.

QoS Requirements of the Control Plane

This section includes the following topics:

- IP Routing
- Network Management

Unless the network is up and running, QoS is irrelevant. Therefore, it is critical to provision QoS for control plane traffic, which includes IP Routing and Network Management traffic.

IP Routing

By default, Cisco IOS software (in accordance with RFC 791 and RFC 2474) marks Interior Gateway Protocol (IGP) traffic such as Routing Information Protocol (RIP/RIPv2), Open Shortest Path First (OSPF), and Enhanced Interior Gateway Routing Protocol (EIGRP) to DSCP CS6. However, Cisco IOS software also has an internal mechanism for granting internal priority to important control datagrams as they are processed within the router. This mechanism is called PAK_PRIORITY.

As datagrams are processed though the router and down to the interfaces, they are internally encapsulated with a small packet header, referred to as the PAKTYPE structure. Within the fields of this internal header there is a PAK_PRIORITY flag that indicates the relative importance of control packets to the internal processing systems of the router. PAK_PRIORITY designation is a critical internal Cisco IOS software operation and, as such, is not administratively configurable in any way.

Note that Exterior Gateway Protocol (EGP) traffic such as Border Gateway Protocol (BGP) traffic is marked by default to DSCP CS6, but does not receive such PAK_PRIORITY preferential treatment and may need to be explicitly protected to maintain peering sessions.

When addressing the QoS needs of IP Routing traffic, Cisco recommends the following guidelines:

- IP Routing traffic should be marked to DSCP CS6; this is default behavior on Cisco IOS platforms.
- IGPs are usually adequately protected with the Cisco IOS internal PAK_PRIORITY mechanism; Cisco recommends that EGPs such as BGP have an explicit class for IP Routing with a minimal bandwidth guarantee.
- Cisco IOS automatically marks IP Routing traffic to DSCP CS6.

Additional information on PAK_PRIORITY can be found at the following URL: http://www.cisco.com/warp/public/105/rtgupdates.html.

Network Management

When addressing the QoS needs of Network Management traffic, Cisco recommends the following guidelines:

- Network Management traffic should be marked to DSCP CS2.
- Network Management applications should be explicitly protected with a minimal bandwidth guarantee.

Network Management traffic is important to perform trend and capacity analysis and troubleshooting. Therefore, you can provision a separate minimal bandwidth queue for Network Management traffic, which could include SNMP, NTP, Syslog, NFS, and other management applications.

Scavenger Class QoS

The Scavenger class, based on an Internet-II draft, is intended to provide deferential services, or "less-than-Best-Effort" services, to certain applications. Applications assigned to this class have little or no contribution to the organizational objectives of the enterprise and are typically entertainment-oriented. These include peer-to-peer (P2P) media-sharing applications (such as KaZaa, Morpheus, Grokster, Napster, iMesh, and so on), gaming applications (Doom, Quake, Unreal Tournament, and so on), and any entertainment video applications.

Assigning a minimal bandwidth queue to Scavenger traffic forces it to be squelched to virtually nothing during periods of congestion, but allows it to be available if bandwidth is not being used for business purposes, such as might occur during off-peak hours. This allows for a flexible, non-stringent policy control of non-business applications.

When provisioning for Scavenger traffic, Cisco recommends the following guidelines:

- Scavenger traffic should be marked to DSCP CS1.
- Scavenger traffic should be assigned the lowest configurable queuing service; for instance, in Cisco IOS this would mean assigning a CBWFQ of 1 percent to Scavenger.

The Scavenger class is a critical component to the data plane policing DoS/worm mitigation strategy presented in the Enterprise QoS SRND 3.1 at www.cisco.com/go/srnd.

Designing the QoS Policies

After a QoS strategy has been defined and the application requirements are understood, end-to-end QoS policies can be designed for each device and interface as determined by its role in the network infrastructure. Because the Cisco QoS toolset provides many QoS design and deployment options, a few succinct design principles can help simplify strategic QoS deployments.

Additionally, these best-practice design principles need to be coupled with topology-specific considerations and constraints. Therefore, the second part of this design section contains a discussion of QoS design considerations specific to the NG-WAN/MAN.

QoS Design Best Practices

For example, one such design principle is to always enable QoS policies in hardware rather than software whenever a choice exists. Lower-end to mid-range Cisco IOS routers (such as the Cisco 1700 through Cisco 7500) perform QoS in software, which places incremental loads on the CPU, depending on the complexity and functionality of the policy. On the other hand, Cisco Catalyst switches and high-end routers (such as the Cisco 7600 and Cisco 12000) perform QoS in dedicated hardware ASICS and as such do not tax their main CPUs to administer QoS policies. This allows complex policies to be applied at line rates at even GE, 10 GE, or higher speeds.

Other simplifying best-practice QoS design principles include the following:

- Classification and marking principles
- Policing and markdown principles
- Queuing and dropping principles

Classification and Marking Design Principles

When classifying and marking traffic, an unofficial differentiated services design principle is to classify and mark applications as close to their sources as technically and administratively feasible. This principle promotes end-to-end differentiated services and per-hop behaviors (PHBs).

Furthermore, it is recommended to use DSCP markings whenever possible, because these are end-to-end, more granular, and more extensible than Layer 2 markings. Layer 2 markings are lost when media changes (such as a LAN-to-WAN/VPN edge). There is also less marking granularity at Layer 2; for example, 802.1Q/p CoS supports only three bits (values 0–7), as does MPLS EXP. Hence only up to eight classes of traffic can be supported at Layer 2 and inter-class relative priority (such as RFC 2597 Assured Forwarding Drop Preference markdown) is not supported. On the other hand, Layer 3 DSCP markings allow for up to 64 classes of traffic, which is more than enough for most enterprise requirements for the foreseeable future.

As the line between enterprises and service providers continues to blur and the need for interoperability and complementary QoS markings is critical, you should follow standards-based DSCP PHB markings to ensure interoperability and future expansion. Because the QoS Baseline marking recommendations are standards-based, enterprises can easily adopt these markings to interface with service provider classes of service. Network mergers (whether the result of acquisitions, mergers, or strategic alliances) are also easier to manage when you use standards-based DSCP markings.

Policing and Markdown Design Principles

There is little reason to forward unwanted traffic only to police and drop it at a subsequent node, especially when the unwanted traffic is the result of DoS or worm attacks. The overwhelming volume of traffic that such attacks can create can cause network outages by driving network device processors to their maximum levels. Therefore, you should police traffic flows as close to their sources as possible.

Whenever supported, markdown should be done according to standards-based rules, such as RFC 2597 (“Assured Forwarding PHB Group”). For example, excess traffic marked to AFx1 should be marked down to AFx2 (or AFx3 whenever dual-rate policing such as defined in RFC 2698 is supported). Following such markdowns, congestion management policies, such as DSCP-based WRED, should be configured to drop AFx3 more aggressively than AFx2, which in turn should be dropped more aggressively than AFx1.

Queuing and Dropping Design Principles

Critical applications such as VoIP require service guarantees regardless of network conditions. The only way to provide service guarantees is to enable queuing at any node that has the potential for congestion, regardless of how rarely this may occur. There is simply no other way to guarantee service levels than by enabling queuing wherever a speed mismatch exists.

When provisioning queuing, some best practice rules of thumb also apply. For example, as discussed previously, the Best Effort class is the default class for all data traffic. Only if an application has been selected for preferential/deferential treatment is it removed from the default class. Because many enterprises have several hundred, if not thousands, of data applications running over their networks, you must provision adequate bandwidth for this class as a whole to handle the sheer volume of applications that default to it. Therefore, it is recommended that you reserve at least 25 percent of link bandwidth for the default Best Effort class.

Not only does the Best Effort class of traffic require special bandwidth provisioning consideration, so does the highest class of traffic, sometimes referred to as the Real-time or Strict Priority class (which corresponds to RFC 3246, “An Expedited Forwarding Per-Hop Behavior”). The amount of bandwidth assigned to the Real-time queuing class is variable. However, if you assign too much traffic for Strict Priority queuing, then the overall effect is a dampening of QoS functionality for non-real-time

applications. Remember that the goal of convergence is to enable voice, video, and data to transparently co-exist on a single network. When real-time applications such as voice or interactive video dominate a link (especially a WAN/VPN link), then data applications fluctuate significantly in their response times, destroying the transparency of the converged network.

Extensive testing and customer deployments have shown that a general best queuing practice is to limit the amount of Strict Priority queuing to 33 percent of link capacity. This Strict Priority queuing rule is a conservative and safe design ratio for merging real-time applications with data applications.

Cisco IOS software allows the abstraction (and thus configuration) of multiple Strict Priority LLQs. In such a multiple LLQ context, this design principle applies to the sum of all LLQs to be within one-third of link capacity.

Note This Strict Priority queuing rule (limit to 33 percent) is simply a best practice design recommendation and is not a mandate. There may be cases where specific business objectives cannot be met while holding to this recommendation. In such cases, enterprises must provision according to their detailed requirements and constraints. However, it is important to recognize the tradeoffs involved with over-provisioning Strict Priority traffic and its negative performance impact on non-real-time-application response times.

Whenever a Scavenger queuing class is enabled, it should be assigned a minimal amount of bandwidth. On some platforms, queuing distinctions between Bulk Data and Scavenger traffic flows cannot be made because queuing assignments are determined by CoS values and these applications share the same CoS value of 1. In such cases you can assign the Scavenger/Bulk Data queuing class a bandwidth percentage of 5. If you can uniquely assign Scavenger and Bulk Data to different queues, then you should assign the Scavenger queue a bandwidth percentage of 1.

NG-WAN/MAN QoS Design Considerations

A few NG-WAN/MAN-specific considerations also come into play when drafting QoS designs for this network. Most of these fall under two main headings:

- MPLS DiffServ tunneling modes, which are discussed in this section.
- Platform-specific capabilities/constraints, discussed in Appendix A, “Platform-Specific Capabilities and Constraints.”

To maintain the class-based per hop behavior, we recommend that you implement the 8 class model where ever possible (see Figure 4-1 for reference) within the MPLS MAN. In scenarios where this is not feasible, at least 5 classes should be deployed—Realtime, Critical, Video, Bulk, and Best Effort. This allows the video traffic to be in a separate queue and keeps bulk data separated from critical data.

MPLS DiffServ Tunneling Modes

A marking disparity exists between MPLS VPNs and DiffServ IP networks:

Layer 2 MPLS labels support only 3 bits for marking (referred to as MPLS EXP bits) offering 8 levels of marking options.

Layer 3 IP packets support 6 bits for marking (the Differentiated Services Code Point [DSCP]).

Because of the reduced level of granularity in marking at Layer 2 via MPLS EXP bits, there is a potential for a “loss in translation” as (potentially) 64 levels of marking cannot be faithfully reduced to 8, nor can 64 levels be faithfully re-created from 8.

To address this disparity, RFC 3270, "Multi-Protocol Label Switching (MPLS) Support of Differentiated Services" presents the following three modes to manage the translation and/or preservation (tunneling) of DiffServ over MPLS VPN networks:

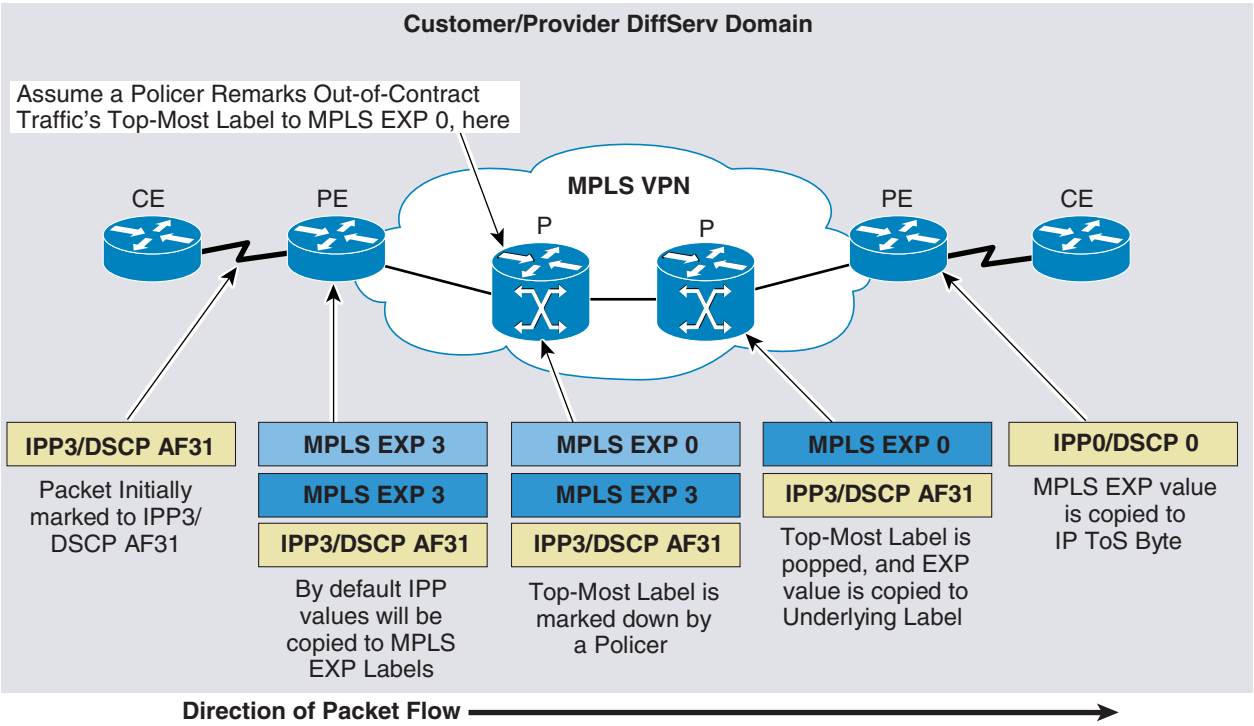
- Uniform mode
- Short-pipe mode
- Pipe mode

Uniform Mode

Uniform mode is used when the customer and service provider share the same DiffServ domain, meaning that a single administrative marking policy is applied over the entire network. In uniform mode, packets are treated uniformly in the IP and MPLS networks; that is, the IP precedence value and the MPLS EXP bits always correspond to the same PHB. Whenever a router changes or recolors the PHB of a packet that change must be propagated to all encapsulation markings. The propagation is performed by a router only when a PHB is added or exposed because of label imposition or disposition on any router in the packet path. The color must be reflected everywhere at all levels. For example, if a packet QoS marking is changed in the MPLS network, the IP QoS marking reflects that change.

Uniform mode is the preferred MPLS DiffServ Tunneling mode for the NG-WAN/MAN and is shown in Figure 4-2.

Figure 4-2 Uniform Mode MPLS DiffServ Tunneling

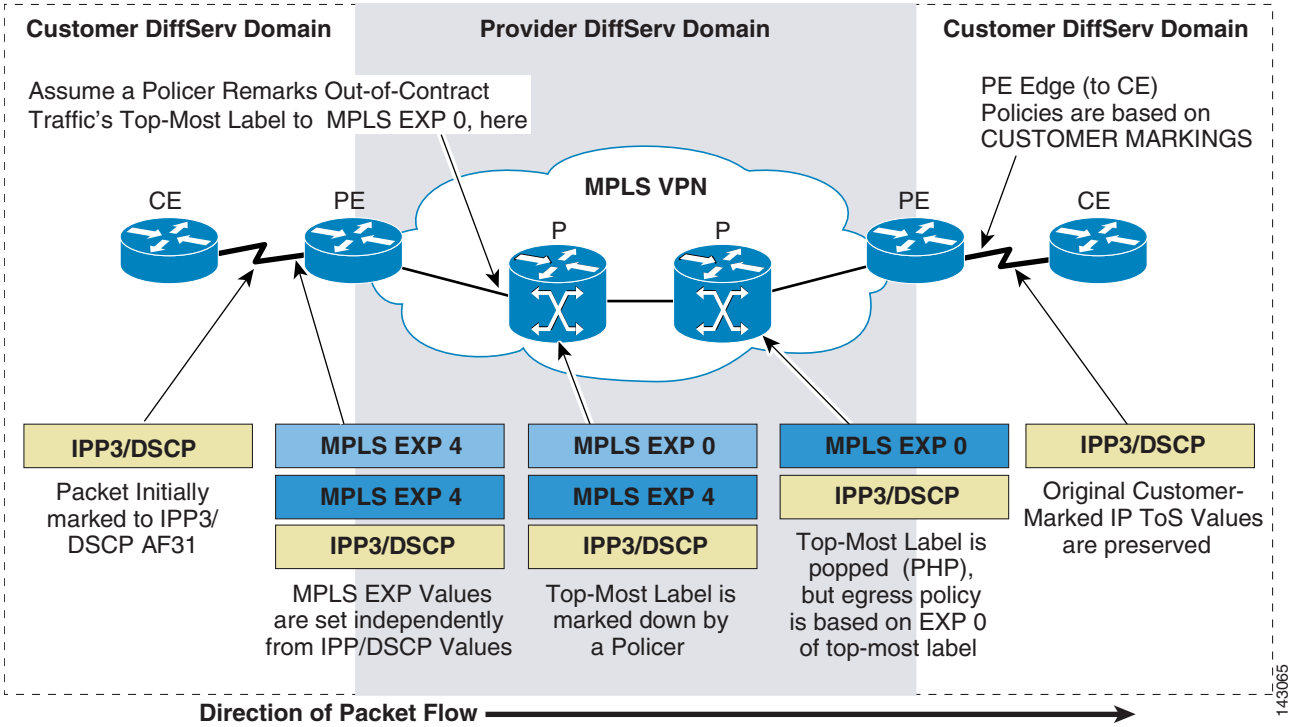


Short-Pipe Mode

Short-pipe mode is used when the customer and service provider are in different DiffServ domains. It allows the service provider to enforce its own DiffServ policy while preserving customer DiffServ information, which provides a DiffServ transparency through the service provider network.

QoS policies implemented in the core do not propagate to the Layer 3 IP packet ToS byte. The classification based on MPLS EXP value ends at the customer-facing egress PE interface; classification at the customer-facing egress PE interface is based on the original IP packet header and not the MPLS header. The presence of an egress IP policy (based on the customer PHB marking and not on the provider PHB marking) automatically implies the short-pipe mode. Short-pipe mode is illustrated in Figure 4-3.

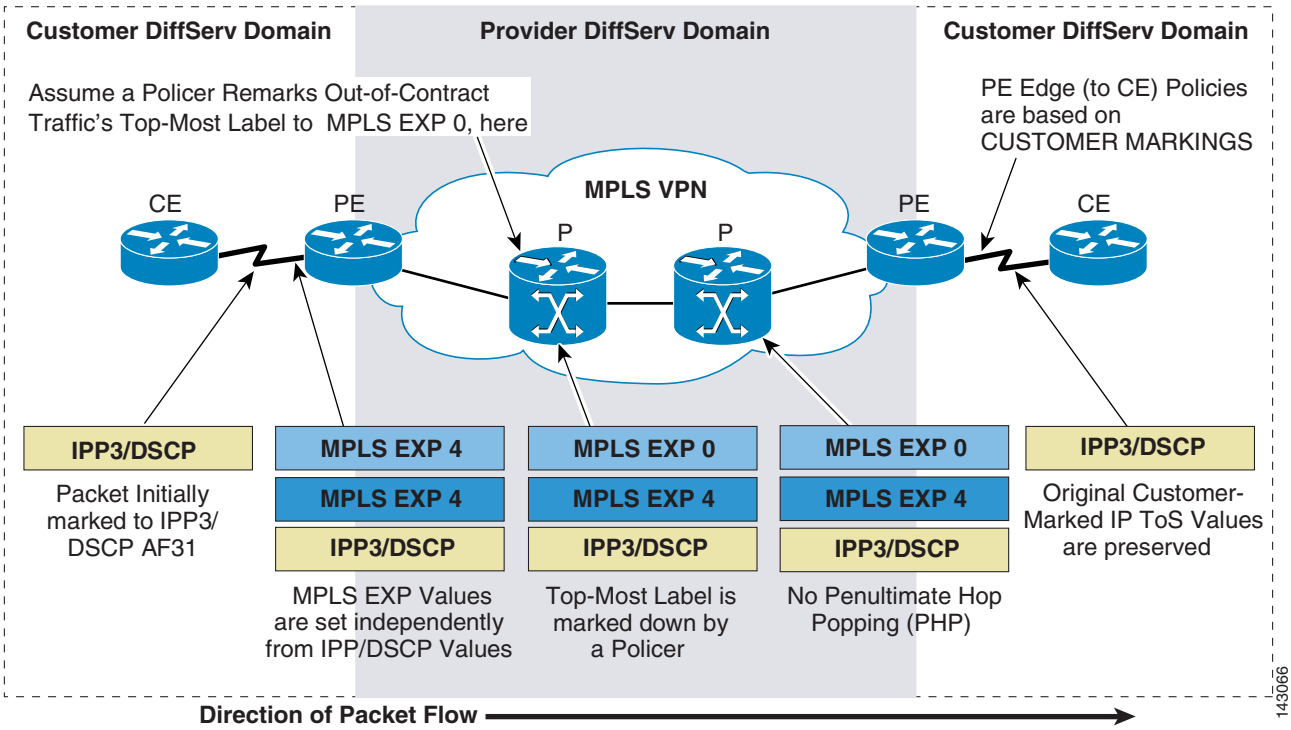
Figure 4-3 Short-Pipe Mode MPLS DiffServ Tunneling



Pipe Mode

Pipe mode is very similar to short-pipe mode, because the customer and service provider are in different DiffServ domains. The difference between the two is that with pipe mode the service provider derives the outbound classification for congestion management and congestion avoidance policies based on the service provider DiffServ policy (rather than according to the enterprise customer markings). This affects how the packet is scheduled/dropped on the egress PE toward the customer CE. Egress scheduling/dropping markings are maintained through the use of **qos-groups** and **discard-class** commands on the egress PE policy maps. This implementation avoids the additional operational overhead of per-customer configurations on each egress interface on the egress PE, as shown in Figure 4-4.

Figure 4-4 Pipe Mode MPLS DiffServ Tunneling



Note Platform-specific design considerations for the Cisco 7200, Cisco 7304,Cisco 7600, and Cisco 12xxx are disccused in Appendix A, "Platform-Specific Capabilities and Constraints."

Security

IP VPNs have a similar security level to that of separate Frame Relay or ATM circuits. Although the separation between the groups is logical, it is very difficult for a hacker to leak traffic from one VPN to another.

In the case of VPNs, isolation is based on the fact that each VPN has a separate logical control plane. This means that devices in one VPN do not know about the IP prefixes in other VPNs and therefore cannot reach these. This protects one VPN from another, but also protects the global routing space from being accessed by users/devices in any of the customer VPNs. By deploying Layer 3 VPNs, the core is therefore made invisible to the customers serviced in the different VPNs, which raises the level of security and availability of the network core.

Encryption

MPLS VPNs provide traffic separation through the logical isolation of the different control planes. MPLS VPNs do not provide encryption of the data in the VPNs. However, the deployment of MPLS VPNs does not preclude encryption of the data in each VPN. Because each VPN provides any-to-any IP connectivity between CE devices, it is possible to overlay many types of encryption architectures on top

of a VPN. Thus, solutions such as DMVPN or site-to-site IPsec can be used to encrypt traffic between CE devices within a VPN. These encryption solutions are a topic in themselves and are discussed in detail in the companion IPsec design guide.

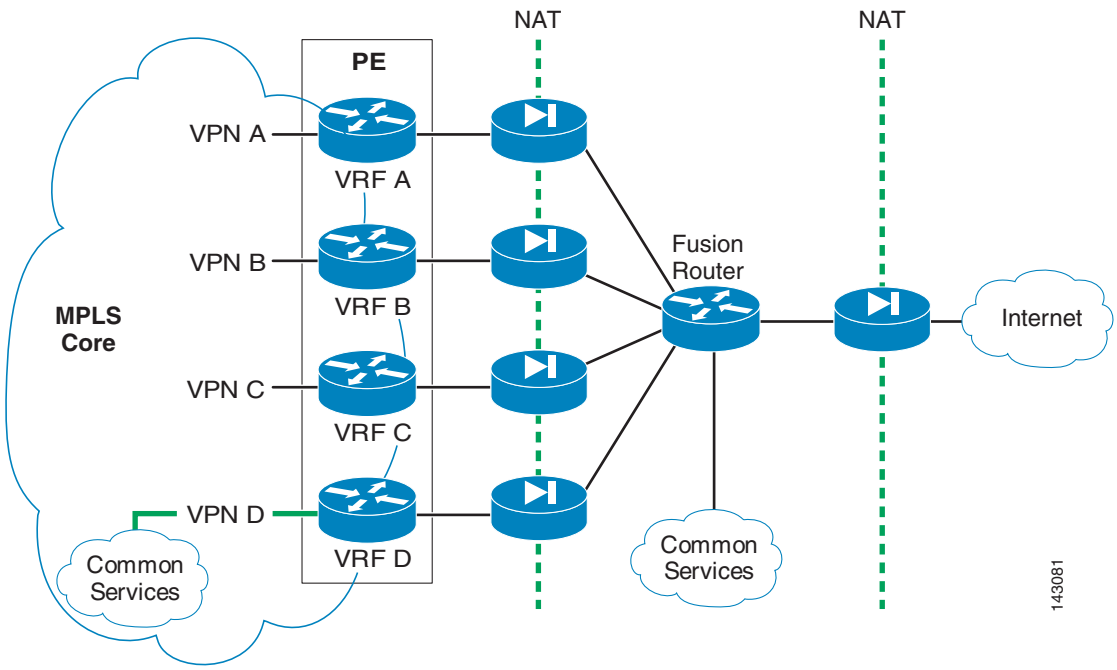
VPN Perimeter—Common Services and the Internet

The default state of a VPN is to be totally isolated from other VPNs. In this respect, VPNs can be seen as physically separate networks. However, because VPNs actually belong to a common physical network, it is desirable for these VPNs to share certain services such as Internet access, DHCP services, DNS services, or server farms. These services are usually located outside of the different VPNs or in a VPN of their own, so these VPNs must have a gateway to connect to the "outside world." The outside world is basically any network outside the VPN such as the Internet or other VPNs. Because this is the perimeter of the VPN, it is also desirable that this perimeter be protected by security devices such as firewalls and IDS. Typically, the perimeter is deployed at a common physical location for most VPNs. Thus, this location is known as the central services site.

The creation of VPNs can be seen as the creation of security zones, each of which has a unique and controlled entry/exit point at the VPN perimeter. Routing within the VPNs should be configured so that traffic is steered to the common services site as required.

Figure 4-5 illustrates a typical perimeter deployment for multiple VPNs accessing common services. Because the services accessed through the VPN perimeter are protected by firewalls, they are referred to as "protected services."

Figure 4-5 Central Site Providing VPN Perimeter Security

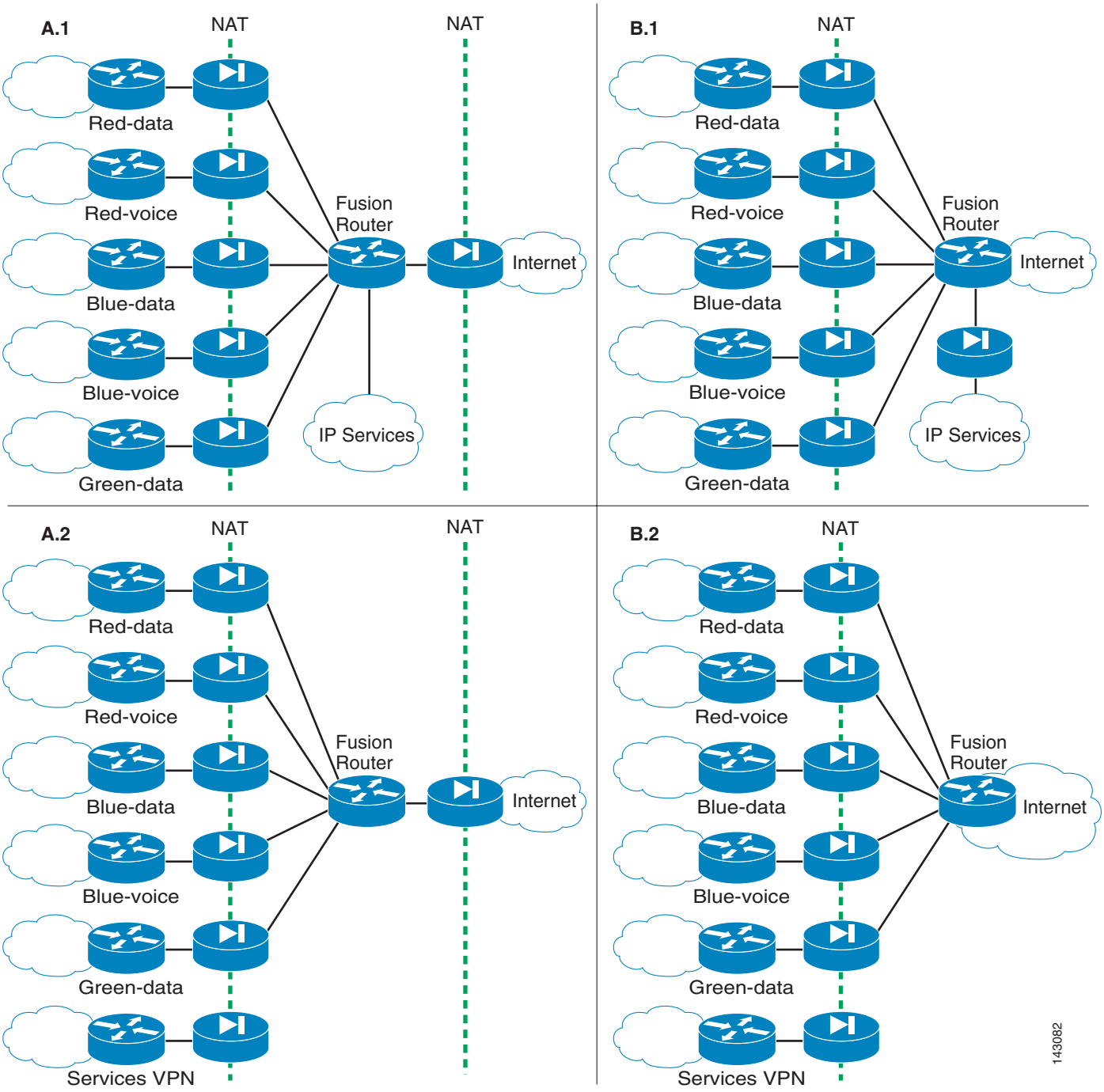


As seen in Figure 4-5, each VPN is head-ended by a dedicated firewall, which allows the creation of security policies specific to each VPN and independent from each other. To access the shared services, all firewalls are connected to a "fusion" router. The fusion router can provide the VPNs with connectivity to the common services, the Internet, or even inter-VPN connectivity. The presence of this fusion router raises two main concerns: the potential for traffic leaking between VPNs and the risk of routes from one VPN being announced to another VPN. The presence of dedicated per VPN firewalls prevents the leaking of traffic between VPNs through the fusion router by allowing only established connections to return through the VPN perimeter. It is important to configure the routing on the fusion device so that routes from one VPN are not advertised to another through the fusion router. The details of the routing configuration at the central site are discussed in Common Services.

Figure 4-5 shows an additional firewall separating the fusion area from the Internet. This firewall is optional. Whether to use it or not depends on the need to keep common services or transit traffic in the fusion area protected from the Internet.

Figure 4-6 illustrates the different scenarios for common services positioning and the Internet firewall.

Figure 4-6 Common Services Positioning



When the common services are not present or placed in their own VPN (and therefore front-ended by a dedicated firewall context), the additional Internet firewall can be removed as shown in diagram B.2. If there is a concern about transit traffic being on the Internet, then the firewall can be kept (see diagram A.2). The common services can be separated from the rest of the network by having their own firewall, yet not be included in a VPN, as shown in diagram B.1.

For scenarios B.1 and B.2, it is important to note that the fusion router is actually part of the Internet and thus the NAT pool employed at the firewalls must use valid Internet addresses. The deployment of the optional Internet firewall should follow standard Internet edge design guidance as documented in the Data Center Internet Edge SRND:

- http://www.cisco.com/application/pdf/en/us/guest/netsol/ns304/c649/ccmigration_09186a008014ee4e.pdf

Throughout this design guide, scenario A.1 is used to illustrate the relevant design and deployment considerations.

Unprotected Services

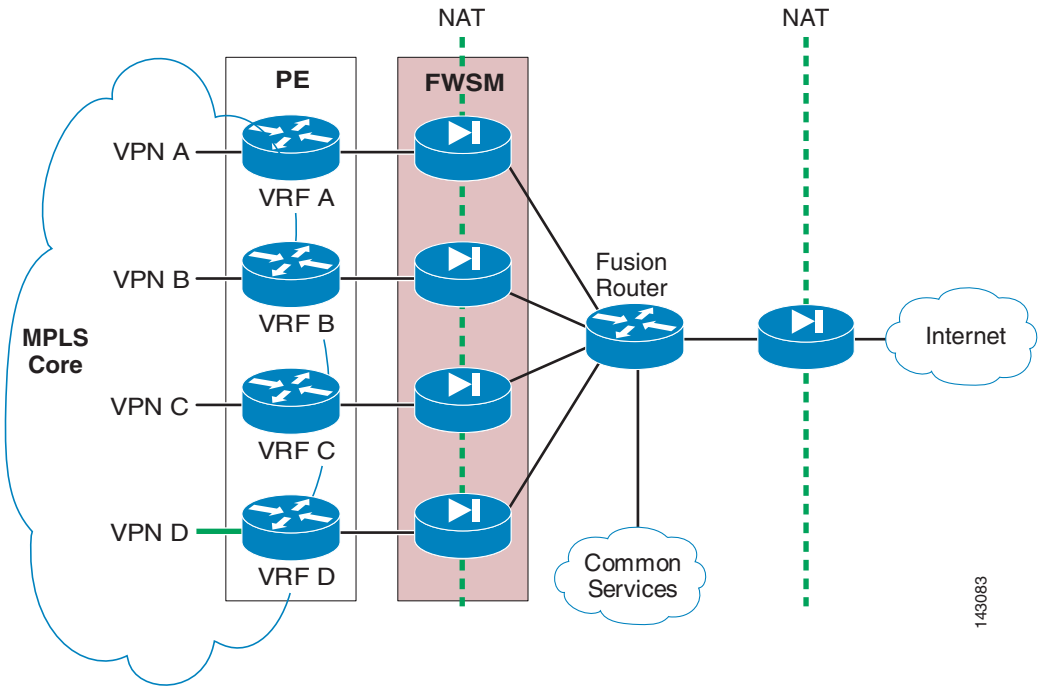
Unlike circuit-based technologies such as ATM or Frame Relay, the IP nature of MPLS VPNs allows enough flexibility for traffic to be leaked between VPNs in a controlled manner by importing and exporting routes between VPNs to provide IP connectivity between the VPNs. Thus, the exchange of traffic between the VPNs may happen within the IP core and does not have to pass through the VPN perimeter firewalls at the central site. This type of inter-VPN connectivity can be used to provide services that do not need to be protected by the central site firewall or that represent an unnecessary burden to the VPN perimeter firewalls. Because of the any-to-any nature of an IP cloud, there is very little chance of controlling inter-VPN traffic after the routes have been exchanged. These are referred to as “unprotected services.” This type of connectivity must be deployed very carefully because it can potentially create unwanted backdoors between VPNs and break the concept of the VPN as a “security zone” protected by a robust VPN perimeter front end. You must also consider the fact that importing and exporting routes between VPNs precludes the use of overlapping address spaces between the VPNs.

Note Although these services are not protected by the VPN perimeter firewalls, the IP segment to which they belong can potentially be head-ended by a firewall and therefore “protected.” However this creates routing and policy management challenges.

Firewalling for Common Services

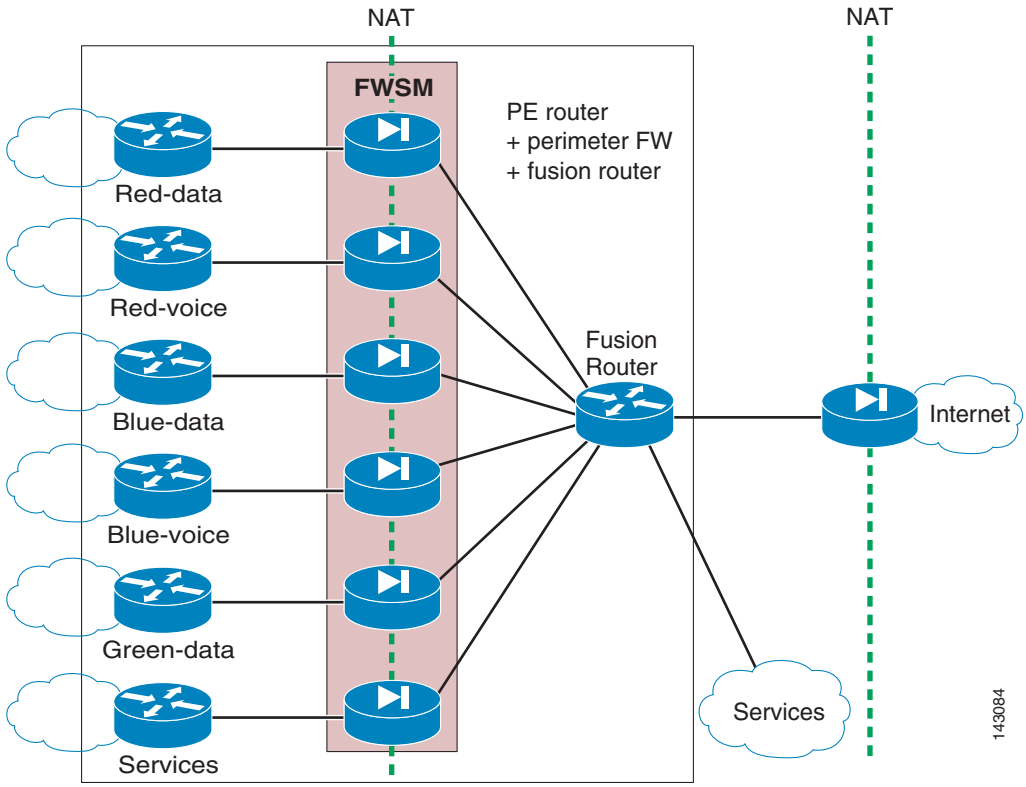
As VPNs proliferate, head-ending each VPN onto its own firewall can become both expensive and hard to manage. Cisco firewalls can be virtualized and thus offer a separate context for each VPN on the same physical appliance. The resulting topology is depicted in Figure 4-7. What has changed here is that a single physical firewall now provides a dedicated logical firewall to each VPN.

Figure 4-7 Virtual Firewall Contexts



The concept of virtual firewalls or firewall contexts has been implemented in the integrated Firewall Service Module (FWSM) for the Cisco Catalyst 6500. The integration of the firewall functionality onto the PE platform allows the topology shown in Figure 4-7 to be consolidated onto a single physical device, as shown in Figure 4-8.

Figure 4-8 Single Box Implementation of the VPN Perimeter Gateway



The logical topology remains unchanged: the firewall functionality is carried out by an FWSM within the PE and the fusion router is implemented by the creation of a VRF inside the same PE. Note that the "fusion VRF" does not connect to the MPLS cloud directly and acts as a separate router, with certain limitations that are explored in a subsequent section.

A single box perimeter implementation is feasible when there is a single common services/Internet site. However, when there is more than one services site and both resiliency and load distribution are desired among those sites, it is necessary to move the fusion VRF outside the PE router and to use a separate physical fusion router. The topologies and necessary routing configuration for single and multiple service site support are discussed in Common Services.

Network Address Translation—NAT

When operating in routed mode, a firewall establishes a connection between the inside and the outside for each flow that traverses the firewall. These connections are in the form of NAT entries, regardless of whether address translation is configured on the firewall.

The default behavior of firewalls is to allow the establishment of flows that are initiated from the inside network. Provided that the access lists allow it, upstream traffic flows through the firewall without a problem. However, a valid NAT entry in the connection table is required for the firewall to allow return traffic through. This NAT entry is dynamically created when the flow is initiated from the inside; connections initiated from the outside do not dynamically create an entry in the firewall.

This unidirectional mechanism prevents connections from being initiated from the outside of the network. For a connection to be successfully initiated from the outside of the network, a NAT entry for the internal destination address must exist in the firewall table before the connection can be established. Thus, if connections initiated from the outside network are required, static NAT entries must be created

to make the specific prefixes available to the outside of the firewall. To allow outside initiated connections, the creation of a static NAT entry is necessary even if the firewall is configured to not translate addresses (nat 0).

Benefits of NAT

The many benefits of being able to translate addresses include:

- Internal networks are hidden from the outside world. With NAT, it is not necessary for the Internet to be aware of the internal addressing scheme of the enterprise to be accessed. This provides an added layer of security.
- Internal networks can use private address spaces as defined in RFC 1918. This is particularly useful when deploying VPNs because this can accelerate the depletion of the IP address space available to the enterprise. This requires restricting extra-VPN communication through the VPN perimeter where addresses can be determined through the use of NAT; that is, inter-VPN route leaking does not work if there are any address overlaps between the private spaces employed.

Dynamic NAT

Address translation can be done dynamically. When an inside station attempts to connect to the outside of the firewall, a dynamic mapping of the source address of the inside station to a globally significant address (outside) is made. The globally significant address to be used is defined by a configured address pool. Thus, each connection is identified by a unique NAT entry. There is the potential for the number of connections to exceed the number of addresses available in the translation pool, in which case any new connection is not successful.

An alternative to regular NAT is Port Address Translation (PAT). With PAT, it is possible to use a single IP address for the global pool. Multiple connections can be associated to the same IP address and are uniquely identified by a unique Layer 4 port number. Hence a single global address can accommodate thousands of connections.

Static NAT

When internal resources must be made available outside the firewall, it is necessary to provide a predictable presence for the internal resource on the outside of the firewall.

By default, all addresses internal to the firewall are not visible to the outside. When addresses are not being translated, they might be visible to the outside but they are still not reachable because reachability from the outside requires an entry in the firewall connection table to be present ahead of time.

Static NAT assigns a globally significant address to the internal resource and also adds an entry to the firewall connection table. This address is fixed so that it can be reached from the outside in a consistent manner. The entry in the connection table makes it possible for the outside to connect to the inside resource provided that the necessary policy is in place.

Common Services

Single Common Services—Internet Edge Site

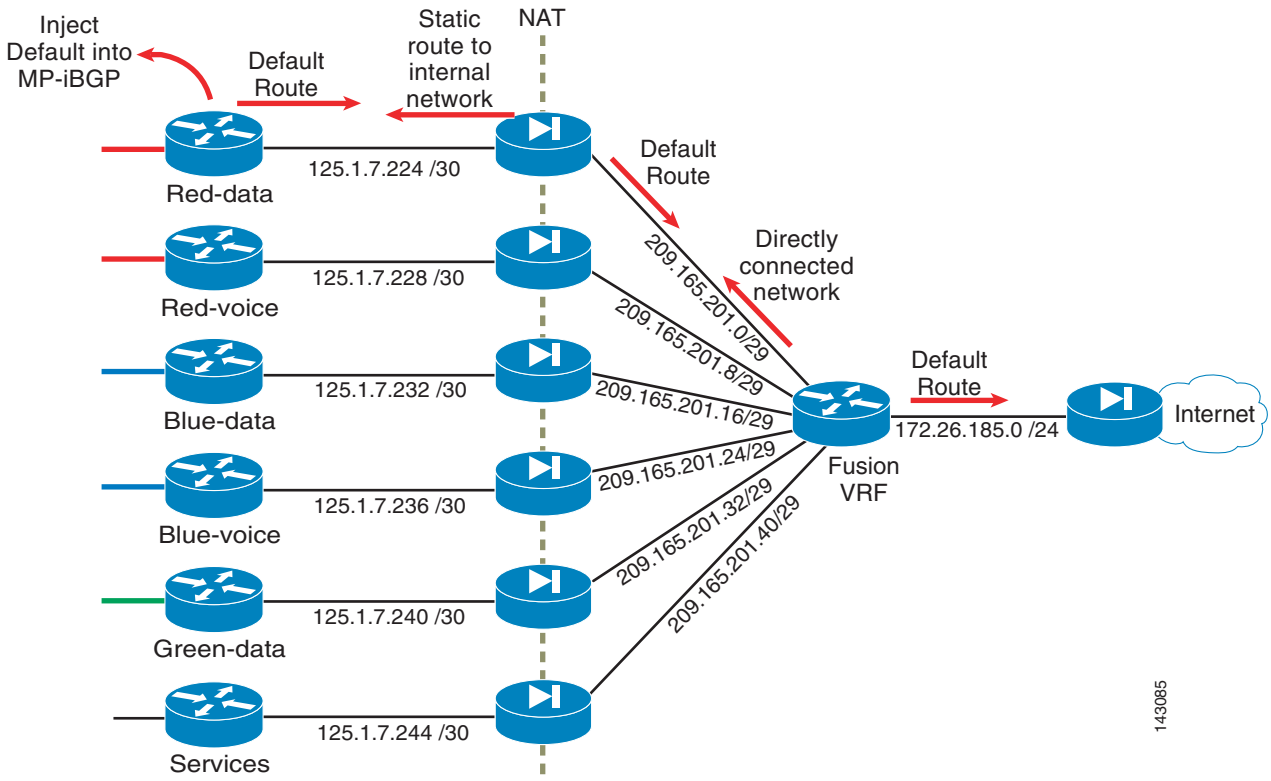
The routing between the fusion router, the different contexts, and VPNs must be configured with care.

Because of its place in the topology, the fusion router has the potential to mix the routes from the different VPNs when exchanging routes dynamically with the different VPNs. However, because the firewall in routed mode supports only static routing when configured for multiple contexts, the mixing

of VPN routes is not a concern. Connectivity between VPNs is achieved by the sole configuration of the fusion router; however the firewalls are configured to allow “established” connections only, which means only connections initiated from the inside of the firewall. Hence all VPNs can reach the fusion router and the fusion router can return traffic to all the VPNs. However, the VPNs are not able to communicate with each other through the fusion router unless very specific policies are set on the different firewall contexts to allow inter-VPN communication through the VPN perimeter gateway.

The static routing configuration for the perimeter gateway is shown in Figure 4-9.

Figure 4-9 Routing Considerations at the VPN Perimeter



The following steps configure static routing for the perimeter gateway. Detail is provided for only one VPN; other VPNs require similar configuration.

Step 1 Create a default route for the internal VRF (red-data):

```
7600-DC1-SS1(config)# ip route vrf red-data 0.0.0.0 0.0.0.0 125.1.7.226
```

Step 2 Create a static route for the inside of the firewall to reach the internal network (red-data VPN):

```
np-fwsm/red-data(config)# route inside 125.1.0.0 255.255.0.0 125.1.7.225 1
```

Step 3 Create a static default route for the outside of the firewall to send traffic to the fusion router/VRF:

```
np-fwsm/red-data(config)# route outside 0.0.0.0 0.0.0.0 209.165.201.2 1
```

Note The fusion router is able to reach the outside prefixes because they are directly connected, so no configuration is required.

Step 4 Create a static default route for the fusion router/VRF to communicate with the ISP. This is the standard configuration of an Internet access router and is not covered in this document.

```
7200-IGATE-DC1(config)# ip route vrf fusion 0.0.0.0 0.0.0.0 172.26.185.1
```

Step 5 Inject the default route created in Step 1 into MP-iBGP:

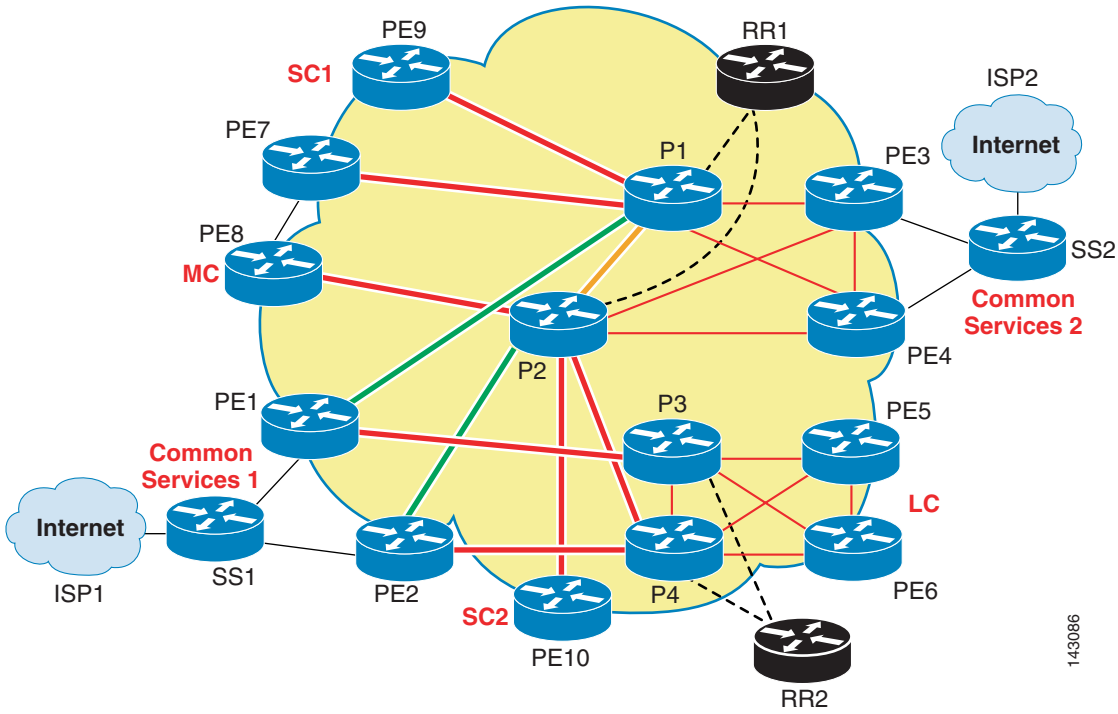
```
7600-DC1-SS1(config)#router bgp 1
7600-DC1-SS1(config-router)#address-family ipv4 vrf red-data
7600-DC1-SS1(config-router-af)#redistribute static
7600-DC1-SS1(config-router-af)#default-information originate

address-family ipv4 vrf red-data
 redistribute connected
 redistribute static
 default-information originate
 no auto-summary
 no synchronization
 exit-address-family
```

Multiple Common Services—Internet Edge Sites

Multiple sites are usually deployed for access to the Internet, so this section focuses on Internet access. However, the same principles apply for other shared services if these are accessible over a common network. When using multiple access points to the Internet (or the common services area), resiliency and load balancing are among the main goals.

In the proposed solution, two common services sites inject a default route into the MPLS MAN. As the default routes are received at the different PEs, the preferred route is chosen by the PE based on its proximity to the common services sites. This proximity is determined based on the core IGP metric (all other BGP attributes should be equal between the two advertised default routes). In the particular case of Internet access, some sites use the first Internet edge site, while others use the second. This achieves site-based load balancing and minimizes the use of the internal MAN links by choosing the closest Internet gateway to send traffic to the Internet in the most efficient manner (see Figure 4-10).

Figure 4-10 Both Internet Edge Sites and IGP Proximity

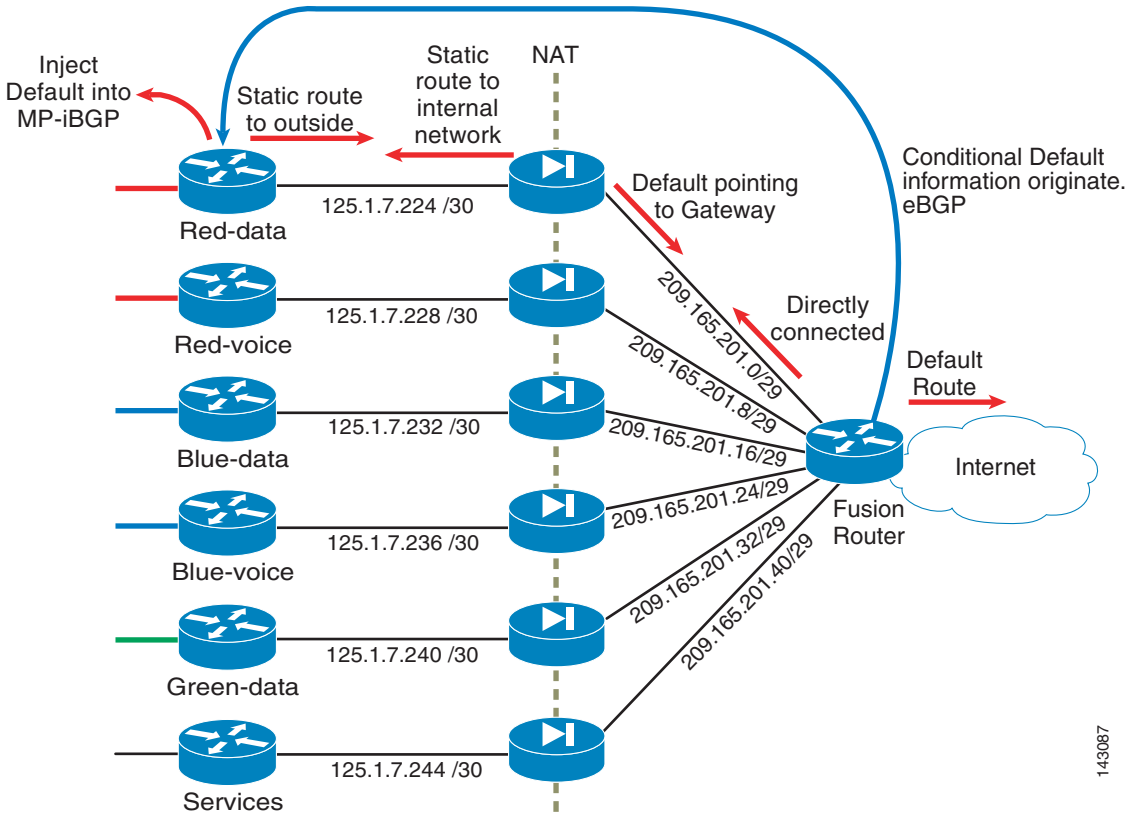


Internet Edge Site Considerations

In the case where one Internet edge site fails, all Internet traffic should be re-routed to the live Internet site. For failures within the MAN, this failover is provided by the reconvergence of the core IGP and the overlaid MP-iBGP. However because static default routes are being injected into the MAN, an ISP failure remains undetected and traffic is black-holed unless there is a dynamic mechanism to report this failure and to trigger a routing re-convergence to use the second Internet edge site. To do this, a dynamic routing protocol can be used to conditionally inject the default routes into the MPLS MAN. Hence a default route is originated and injected into the MPLS MAN from the Internet edge router only if this route is valid; that is, it exists in the edge router table (see Step 9 below).

To achieve this dynamic notification over the perimeter firewalls, eBGP is required to establish a connection across the firewall contexts (contexts do not support dynamic routing protocols). Because eBGP peering is required and this cannot be established between VRFs in a single box (the router ID would be the same for both VRFs, which would prevent the BGP adjacency from being established), a separate physical router is required for the fusion role (see Figure 4-11).

Figure 4-11 EBGP Peering for Dynamic Notification



The following steps must be completed to achieve the necessary BGP peering and inject the default routes conditionally:

Step 1 On the internal VRF, create a static route to the outside firewall subnet (209.165.201.0 /29):

```
7200-IGATE-DC1(config)#ip route vrf red-data 209.165.201.0 255.255.255.248 125.1.7.226
```

Step 2 On the inside firewall interface, create a static route to the internal VPN summary prefix:

```
np-fwsm/red-data(config)#route inside 125.1.0.0 255.255.0.0 125.1.7.225 1
```

Step 3 On the outside firewall interface, create a static default route to the Internet gateway:

```
np-fwsm/red-data(config)#route outside 0.0.0.0 0.0.0.0 209.165.201.2 1
```

Note The fusion router is directly connected to the outside firewall networks. No configuration is required

Step 4 On the fusion router, create a default route pointing at the Internet gateway (172.26.185.1 /32):

```
7200-IGATE-DC1(config)#ip route 0.0.0.0 0.0.0.0 172.26.185.1
```

Step 5 Configure static NAT entries for the internal VRF BGP peering address. These are necessary to establish the bi-directional TCP sessions for BGP peering. For any type of communication to be initiated from the outside of the firewall, a static NAT entry is required by the firewall; otherwise the connection is rejected.

```
static (inside,outside) 209.165.201.3 125.1.7.225 netmask 255.255.255.255 norandomseq
```

Step 6 Open the necessary firewall policies to permit BGP peering over the firewall:

```
access-list allow_any extended permit ip any any log debugging !Allows sessions initiated
from the inside of the firewall (i.e. the VPN).
access-list allow_any extended permit tcp host 125.1.7.225 eq bgp host 209.165.201.2 eq
bgp
access-list allow_bgp extended permit tcp host 209.165.201.2 eq bgp host 209.165.201.3 eq
bgp
!
access-group allow_any in interface inside
access-group allow_bgp in interface outside
```

Step 7 Configure the internal VRFs and the fusion router as BGP neighbors:

```
!Fusion Router!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
router bgp 10
 no synchronization
 bgp log-neighbor-changes
 redistribute static
 neighbor 209.165.201.3 remote-as 1
 neighbor 209.165.201.3 ebgp-multihop 255
 !
!PE router: Red-data VRF!!!!!!!!!!!!!!!!!!!!!!!!!
router bgp 1
 no synchronization
 bgp log-neighbor-changes
 neighbor 125.1.125.15 remote-as 1
 neighbor 125.1.125.15 update-source Loopback0
 neighbor 125.1.125.16 remote-as 1
 neighbor 125.1.125.16 update-source Loopback0
 neighbor 209.165.201.2 remote-as 10
 no auto-summary
 !
 address-family vpnv4
 neighbor 125.1.125.15 activate
 neighbor 125.1.125.15 send-community extended
 neighbor 125.1.125.16 activate
 neighbor 125.1.125.16 send-community extended
 exit-address-family
 !
 address-family ipv4 vrf red-data
 redistribute connected
 redistribute static
 neighbor 209.165.201.2 remote-as 10
 neighbor 209.165.201.2 ebgp-multihop 255
 neighbor 209.165.201.2 activate
 maximum-paths eibgp 2
 no auto-summary
 no synchronization
 exit-address-family
 !
```

Step 8 Originate a default route at the fusion router and send it over BGP to the internal VRFs. Use conditional statements so that the default route is advertised only if it is present in the local routing table (that is, if the Internet service is available).

```
router bgp 10
 no synchronization
 bgp log-neighbor-changes
 redistribute static
 neighbor 209.165.201.3 remote-as 1
 neighbor 209.165.201.3 ebgp-multihop 255
 neighbor 209.165.201.3 default-originate route-map SEND_DEFAULT
 neighbor 209.165.201.3 distribute-list 3 in
```

```
 no auto-summary
!
ip classless
ip route 0.0.0.0 0.0.0.0 172.26.185.1
no ip http server
!
!
access-list 1 permit 0.0.0.0
access-list 2 permit 172.26.185.1
access-list 3 deny   any
!
route-map SEND_DEFAULT permit 10
 match ip address 1
 match ip next-hop 2
 set metric 0
 set local-preference 100
```

Step 9 Prevent any BGP updates from the inside network coming onto the fusion router. If the fusion router is allowed to receive VPN routes via e-BGP, it replicates the received routes onto its other e-BGP peers. This would basically inject routes from one VPN into another, so these updates must be prevented.

```
router bgp 10
 no synchronization
 bgp log-neighbor-changes
 redistribute static
 neighbor 209.165.201.3 remote-as 1
 neighbor 209.165.201.3 ebgp-multihop 255
 neighbor 209.165.201.3 default-originate route-map SEND_DEFAULT
 neighbor 209.165.201.3 distribute-list 3 in
 no auto-summary
!
ip classless
ip route 0.0.0.0 0.0.0.0 172.26.185.1
no ip http server
!
!
access-list 1 permit 0.0.0.0
access-list 2 permit 172.26.185.1
access-list 3 deny   any
!
```

Routing Considerations

Advertising Multiple Routes into MP-iBGP

Advertising more than one default route or advertising multiple routes for the same prefix must be done with care. The default behavior of a route reflector is to make a decision based on metrics and attributes and to reflect only the best one of the advertised routes. The result is that all PEs always receive the route that is best for the route reflector, which is not necessarily the best route for the PE to reach the Internet.

To achieve load balancing and redundancy from injecting multiple routes for a common destination in this topology, it is important that the route reflector actually "reflects" all the routes it receives so that the route selection can actually be done at the PEs. To achieve this, the routes must be advertised with different RDs. For example, the default route advertised by Common Services Site 1 (SS1) is sent with an RD of 10:103, while the default route sent by Common Services Site 2 (SS2) is sent with an RD of 101:103. In this manner, some sites prefer SS2 while others prefer SS1.

Load balancing across the MAN core can be achieved by instructing BGP to install multiple paths in the routing table (**ibgp multipath**). Although it is tempting to use unequal cost paths and to load balance across all possible paths, this may affect the way traffic to the Internet is handled and may cause the use of suboptimal paths to access the Internet. In the proposed scenario, the requirement is for certain portions of the network to prefer on Common Services Site over another. Thus the load balancing is done per site rather than per flow. For example, site SC1 always tries to use SS1 first because it is the closest Internet access site. If unequal paths are allowed to be installed in the routing table, SC1 sends some flows over SS1 and others over SS2, potentially congesting low speed links that would not have been used if only one path had been installed on the routing table.

However, the solution is not to turn **bgp multipath** off, but to set the bgp multipath capability to install only multiple equal cost paths. This is important because equal cost load balancing is desirable between sites. Because only equal cost paths can be installed in the table, the Internet is accessed consistently via either SS1 or SS2, depending on the proximity of the site. If a failure is detected, the routing protocols must determine which sites are still available and recalculate the paths to the Common Services Sites to make a decision on where to exit the Internet.

Asymmetric Return Paths

This is a classic problem faced when multi-homing to the Internet, in which traffic exits the enterprise out of one gateway and the return traffic is received over a different gateway. The implications are many, but the main one is that the return traffic is normally not able to get through the firewall; no session has been established at the return firewall because the traffic originally left the network through a different firewall.

In the proposed scenario, the asymmetry of the return path is handled by using different global NAT address pools outside the different Internet gateways. Each Internet gateway advertises a unique address pool, thus eliminating any ambiguity in the return path. For example, the source address of traffic leaving SS1 is rewritten to a prefix advertised only by SS1. Therefore the return traffic for a stream that entered the Internet through SS1 must be through SS1 because the Internet has routes to the SS1 address pool only through SS1.

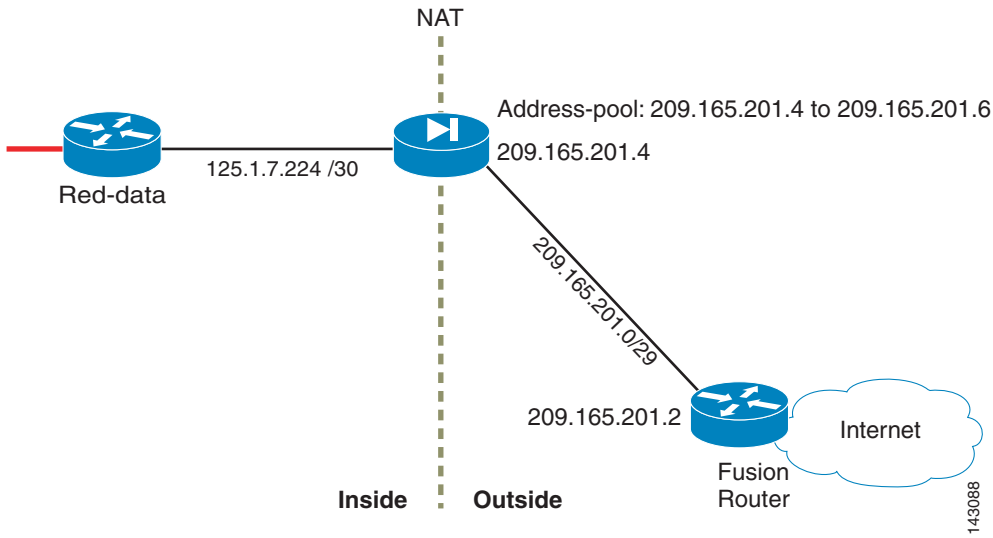
NAT in the MPLS MAN

A combination of dynamic and static NAT is required at the VPN perimeter:

- Dynamic NAT is used to allow connectivity for sessions established from the inside of the network.
- Static NAT is required to allow the following:
  - BGP peer establishment
  - Connectivity to resources shared from inside a service VPN

Dynamic NAT can be established by using either NAT or PAT. When using NAT, it is necessary to provide the outside interface of the firewall with an IP prefix that can accommodate the entire global address pool to be used in the translation. Figure 4-12 shows the scenario for the Red-data VPN in the Cisco test bed.

Figure 4-12 Red-data VPN Test Bed Scenario



Any connection from the Red-data VPN to the Internet creates one NAT entry and therefore uses one of the addresses in the address pool. Thus, the number of possible concurrent connections is limited to three in this specific scenario. Note that a 29-bit address mask (rather than 32 bits) has been used for the point-to-point connection to accommodate the NAT address pool.

The following commands configure the Red-data firewall context to allow this kind of connectivity:

```
! Create the dynamic NAT Pool
global (outside) 1 209.165.201.4-209.165.201.6 netmask 255.255.255.248
nat (inside) 1 125.1.0.0 255.255.0.0
```

The following commands allow outbound connectivity:

```
!Allow sessions initiated from the inside of the firewall (i.e. the VPN).
access-list allow_any extended permit ip any any log debugging
access-group allow_any in interface inside
```

Alternatively, PAT can provide dynamic translation without the limitation of the exhaustion of the global address pool. Configuring PAT is almost identical to configuring NAT, except that instead of defining a global range, a single IP is configured:

```
! Create the dynamic PAT Pool
np-fwsm/red-data(config)# nat (inside) 1 125.1.0.0 255.255.0.0
np-fwsm/red-data(config)# global (outside) 1 209.165.201.4
Global 209.165.201.4 will be Port Address Translated
```

A static NAT entry is required to allow BGP peering between the fusion router and the internal VRF as described in Firewalling for Common Services.

The necessary access lists must be configured to allow this type of connectivity as well. Care must be taken to open the firewall exclusively to the relevant BGP traffic, as shown in the following configuration:

```
! Create the static translation for the inside (125.1.7.225) peer
static (inside,outside) 209.165.201.3 125.1.7.225 netmask 255.255.255.255 norandomseq
! Allow bgp tcp session between the neighbors only and in both directions
access-list allow_any extended permit tcp host 125.1.7.225 eq bgp host 209.165.201.2 eq
bgp
access-list allow_bgp extended permit tcp host 209.165.201.2 eq bgp host 209.165.201.3 eq
bgp
! Apply policies in both directions
```

```
access-group allow_any in interface inside
access-group allow_bgp in interface outside
```

Other static NAT entries may be required if there are servers inside the VPN that are made available outside the VPN. As the number of servers to publish increases, the use of static PAT may be useful. The use of static PAT is beyond the scope of this document; for information on static PAT as well as more details on NAT in general, see the FWSM configuration guide at: http://www.cisco.com/univercd/cc/td/doc/product/lan/cat6000/mod_icn/fwsm/fwsm_2_2/fwsm_cfg/index.htm

Convergence

Redundancy within the MPLS network—power supplies, links, routers, etc.—provides protection against any such losses. But more and more applications require faster convergence which from a network perspective involves:

- Detecting the failure
- Finding an alternate resource or restoration
- Propagation of the change to the rest of the network, if required

The commonly used mechanisms in IP-environments dictates that an IGP extended for Fast Convergence together with convergence enhancements for BGP provides the overall protection at restoration function. Traditional MPLS Protection and Restoration mechanisms, such as Traffic Engineering Fast ReRoute (TE FRR), provide capabilities to circumvent node or link failures.

- **Link failure detection**—Various mechanisms are in place to provide a fast detection of link failure, both generic and media dependent. The fastest mechanism by far is the integrated OAM mechanism of SONET/SDH framing. Other mechanisms include Loss of optical Signal (LOS), PPP keepalives, and various LMI mechanisms. Bidirectional Forwarding Detection (BFD) is a generic lightweight hello-based mechanism that can be used in conjunction with any type of media.
- **Failure propagation**—Depending on the Protection and Restoration mechanism being used, there may not be an associated propagation delay before the backup for a failed facility is installed. This is the case with TE FRR. If an IGP or BGP is used then the updated network information has to be flooded throughout the network.

Additional processing time is required for the IGP to compute a new network view by performing an SPF operation. Once that operation is completed, updated routing information is installed in the RIB. In an MPLS network protected by TE FRR, this operation still takes place, but the service restoration is not dependent on its completion. In the case of BGP, an update or bestpath operation has to be performed and the time this operation takes is a direct consequence of the BGP table size. After the RIB has been updated the associated FIB also has to be updated so that the forwarding plane can make use of the updated information.

Traffic Engineering Fast ReRoute (TE FRR)

FRR is a mechanism for protecting MPLS TE LSPs from link and node failures by locally repairing the LSPs at the point of failure, allowing data to continue to flow on them while their headend routers attempt to establish new end-to-end LSPs to replace them. FRR locally repairs the protected LSPs by rerouting them over backup tunnels that bypass failed links or nodes.

- **Link Protection**—Backup tunnels that bypass only a single link of the LSP's path provide link protection. They protect LSPs if a link along their path fails by rerouting the LSP's traffic to the next hop (bypassing the failed link). These are referred to as next-hop (NHOP) backup tunnels because they terminate at the LSP's next hop beyond the point of failure.
- **Node Protection**—Backup tunnels that bypass next-hop nodes along LSP paths are called next-next-hop (NNHOP) backup tunnels because they terminate at the node following the next-hop node of the LSP paths, thereby bypassing the next-hop node. They protect LSPs if a node along their path fails by enabling the node upstream of the failure to reroute the LSPs and their traffic around the failed node to the next-next hop.
- **RSVP Hellos**—RSVP Hello enables RSVP nodes to detect when a neighboring node is not reachable. This provides node-to-node failure detection. When such a failure is detected, it is handled in a similar manner as a link-layer communication failure.

RSVP Hello can be used by FRR when notification of link-layer failures is not available (for example, with Ethernet) or when the failure detection mechanisms provided by the link layer are not sufficient for the timely detection of node failures.

A node running Hello sends a Hello Request to a neighboring node every interval. If the receiving node is running Hello, it responds with Hello Ack. If four intervals pass and the sending node has not received an Ack or it receives a bad message, the sending node declares that the neighbor is down and notifies FRR. There are two configurable parameters:

- Hello interval—Use the **ip rsvp signalling hello refresh interval** command.
- Number of acknowledgment messages that are missed before the sending node declares that the neighbor is down—Use the **ip rsvp signalling hello refresh misses** command.

Fast Reroute Activation

There are two mechanisms that cause routers to switch LSPs onto their backup tunnels:

- Interface down notification
- RSVP Hello neighbor down notification

When a router's link or neighboring node fails, the router often detects this failure by an interface down notification. For example, on a POS interface this notification is very fast. When a router notices that an interface has gone down, it switches LPSs going out that interface onto their respective backup tunnels.

RSVP Hellos can also be used to trigger FRR. If RSVP Hellos are configured on an interface, messages are periodically sent to the neighboring router. If no response is received, Hellos declare that the neighbor is down. This causes any LSPs going out that interface to be switched to their respective backup tunnels.

An additional mechanism that will be available in the future would be BFD. BFD is a detection protocol that is designed to provide fast forwarding path failure detection times for all media types, encapsulations, topologies, and routing protocols. In addition to fast forwarding path failure detection, BFD provides a consistent failure detection method for network administrators. Because the network administrator can use BFD to detect forwarding path failures at a uniform rate, rather than the variable rates for different routing protocol hello mechanisms, network profiling and planning will be easier, and reconvergence time will be consistent and predictable.

As long as each BFD peer receives a BFD control packet within the detect-timer period, the BFD session remains up and any routing protocol associated with BFD maintains its adjacencies. If a BFD peer does not receive a control packet within the detect interval, it informs any clients of that BFD session about the failure.

Note As of August, 2006, BFD support for FRR triggering is planned to 12.0(33)S for GSR and the "cobra" release for 7600.

Backup Tunnel Selection Procedure

When an LSP is signaled, each node along the LSP path that provides FRR protection for the LSP selects a backup tunnel for the LSP to use if either of the following events occurs:

- The link to the next hop fails.
- The next hop fails.

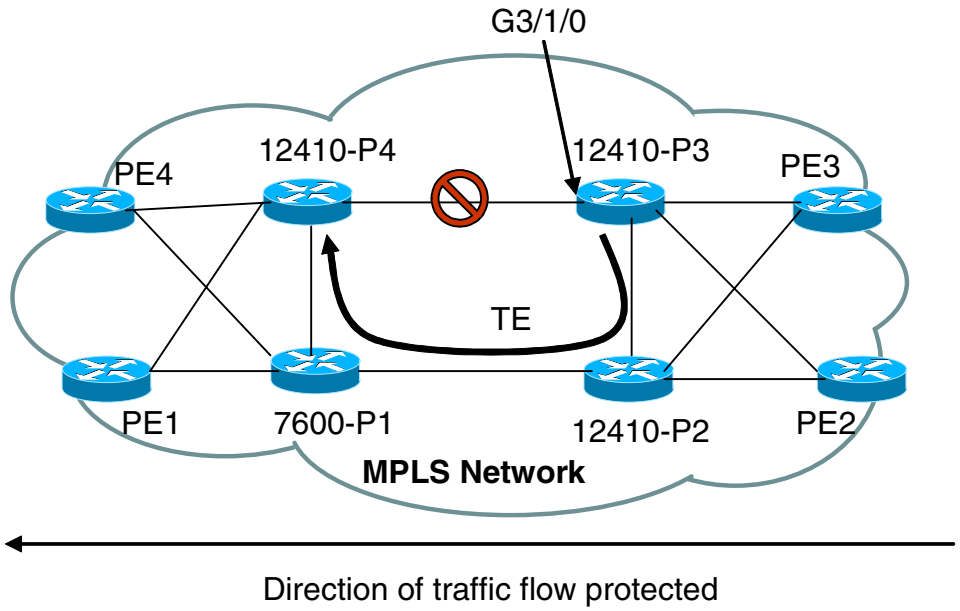
By having the node select the backup tunnel for an LSP before a failure occurs, the LSP can be rerouted onto the backup tunnel quickly if there is a failure.

For an LSP to be mapped to a backup tunnel, all of the following conditions must exist:

- The LSP is protected by FRR; that is, the LSP is configured with the **tunnel mpls traffic-eng fast-reroute** command.
- The backup tunnel is up.
- The backup tunnel is configured to have an IP address, typically a loopback address.
- The backup tunnel is configured to protect this LSP's outgoing interface; that is, the interface is configured with the **mpls traffic-eng backup-path** command.
- The backup tunnel does not traverse the LSP's protected interface.
- The backup tunnel terminates at the LSP's NHOP or NNHOP. If it is an NNHOP tunnel, it does not traverse the LSP's NHOP.
- The bandwidth protection requirements and constraints, if any, for the LSP and backup tunnel are met.

Protecting the Core Links

Figure 4-13 Protecting the Core Links



In Figure 4-13, the link between P3 and P4 is protected (for traffic going from P3 to P4) by a explicitly configured TE tunnel P3-P2-P1-P4. Since the tunnels are unidirectional, a reverse path would need to be created for traffic going from P4 to P3. All the links are POS and rely on underlying SONET alarms for link failure detection. Explicit tunnels can be setup to provide fast re-route capabilities in case of core failures.

```
interface Tunnel11
 ip unnumbered Loopback0
 no ip directed-broadcast
 tunnel destination 100.0.250.14
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng priority 0 0
 tunnel mpls traffic-eng bandwidth 10000
 tunnel mpls traffic-eng path-option 5 explicit name backup_of_10
!
interface POS3/1/0
 ip address 100.0.4.1 255.255.255.0
 no ip directed-broadcast
 ip pim sparse-mode
 no keepalive
 mpls traffic-eng tunnels
 mpls traffic-eng backup-path Tunnel11
 tag-switching ip
 crc 32
 clock source internal
 pos ais-shut
 pos report lrdi
 ip rsvp bandwidth 1866000 1866000
!
ip explicit-path name backup_of_10 enable
 next-address 100.0.3.1<P2>
 next-address 100.0.2.11<P1>
 next-address 100.0.1.2<P4>
```

Performance

Failure detection plays an important role in determining the switchover performance. Depending on the platform/linecard/IOS, different detection mechanisms can be deployed, such as SONET alarms for POS, RSVP hellos for other interface types, and potentially BFD in the future. Another factor is how quickly the database get updated once the failure has been signaled, which is dependent on the number of routes being protected.

In the above example with 2000 IGP routes in the network, the failover testing was done to compare the responses with and without TE/FRR. A single stream of traffic was observed end-to-end. When the POS link between P3-P4 was shutdown, the packet loss was measured and the downtime was calculated based on that. It was found that with FRR a POS failure created a failure of 4-5s, but with FRR this was measured to be less then 10ms.

CHAPTER 5

Management

Cisco IP Solution Center (ISC) is a family of intelligent element management applications that help reduce overall administration and management costs by providing automated resource management and rapid profile-based provisioning capabilities. ISC enables fast deployment and time to market of Multiprotocol Label Switching (MPLS) and Metro Ethernet technologies. Cisco ISC 4.0 contains three applications that can operate alone or as a suite in an MPLS Management Solution.

The Cisco ISC MPLS VPN Management (ISC:MPLS) application helps enterprises offering MPLS VPN services by providing the provisioning, planning, and troubleshooting features essential to manage the entire life cycle of MPLS VPN services. MPLS management features include policy-based VPN, management VPN, QoS provisioning, and MPLS VPN routing audit. These features help to guarantee accurate service deployment and to reduce the cost of deploying MPLS VPN services.

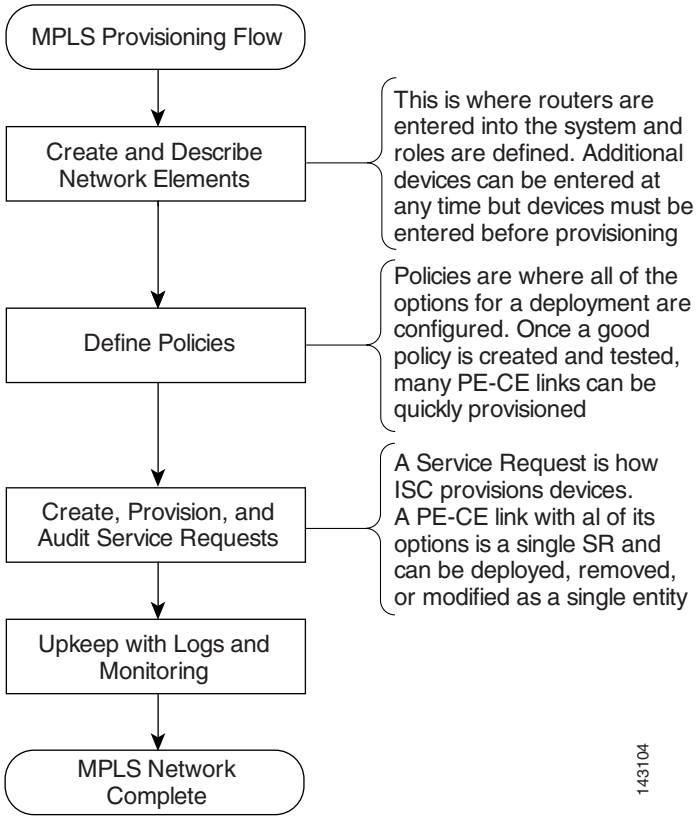
Cisco ISC contains the following competitive advantages:

- Tracking of Layer 3 and Layer 2 resources—Automation of resource management reduces cost of manual and time-consuming tasks and helps ensure accuracy.
- Rapid profile-based provisioning—Helps control operational costs by providing rapid deployment of services.
- Recognizing incorrect service configuration—Reduces the time it takes to troubleshoot network outages because of incorrect service configuration.
- Investment protection from Cisco IOS Software and line card changes—Reduces time to market of new services and lowers the cost of upgrading customer OSS systems because of upgrades in platforms, software versions, and line cards.
- Bandwidth protection planning—Provides a cost-effective alternative to lower-layer protection. Highly efficient use of bandwidth allows more traffic to be supported on the network without compromising protection requirements.
- Resource management—Management of resources such as autonomous system, regions, IP address pools, and provider administrative domains.
- Wide range of supported protocols—Download and activation of the Layer 3 VPN service design, including activation of various MPLS VPN topologies and wide support for routing protocol configuration on attachment circuits for export of customer routes: Open Shortest Path First (OSPF), static, Enhanced Interior Gateway Routing Protocol (EIGRP), Intermediate System-to-Intermediate System (IS-IS), and so on.
- Pre-deployment design verification—Pre-provisioning checks for validity of service design, including uploading of the current configuration and validation of service design against the existing network configuration.
- Post-deployment verification—Post-provisioning validation of the service design to determine whether the Layer 3 VPN is active and functional.

- Continuing verification—Smart configuration and routing audits and VPN routing and forwarding (VRF) pings to validate VPN configuration and on-demand and scheduled audits for configuration troubleshooting.

Figure 5-1 shows the various components of the management system.

Figure 5-1 Management Flowchart



Related Documents

- IP Solution Center (ISC)—http://www.cisco.com/en/US/products/sw/netmgtsw/ps4748/index.html
- Cisco Info Center (CIC)—http://www.cisco.com/en/US/products/sw/netmgtsw/ps996/index.html

CHAPTER 6

Advanced Applications Over MPLS-Based VPNs

Cisco IP Communications

This section highlights and describes the design differences and requirements of integrating Cisco IP Communications with the self-managed MPLS MAN. It is not intended to provide full details of general enterprise IP Communications design, which is highly complex.

Much of the content in this section that specifically applies to the self-managed MPLS MAN has been taken from the Cisco Enterprise IP Telephony SRND for CallManager 4.1 and the Enterprise QoS Solution Reference Network Design Guide.

See the following URLs for complete details of general deployment design guidance for IP Communications:

- Enterprise IP telephony SRND—http://www.cisco.com/en/US/products/sw/voicesw/ps556/products_implementation_design_guide_book09186a008044714e.html
- QoS SRND — http://www.cisco.com/univercd/cc/td/doc/solution/esm/qossrnd.pdf

Overview of Cisco IP Communications Solutions

Cisco IP Communications solutions deliver fully integrated communications by enabling data, voice, and video to be transmitted over a single network infrastructure using standards-based IP. Leveraging the framework provided by Cisco IP hardware and software products, Cisco IP Communications solutions deliver unparalleled performance and capabilities to address current and emerging communications needs in the enterprise environment. Cisco IP Communications solutions are designed to optimize feature functionality, reduce configuration and maintenance requirements, and provide interoperability with a wide variety of other applications. Cisco IP Communications solutions provide this capability while maintaining a high level of availability, QoS, and security for the network.

Cisco IP Communications encompass the following solutions:

- IP telephony—Transmits voice communications over the network using IP standards. The Cisco IP Telephony solution includes a wide array of hardware and software products such as call processing agents, IP phones, video devices, and special applications.
- Unified communications—Delivers powerful unified messaging (email, voice, and fax messages managed from a single inbox) and intelligent voice messaging (full-featured voicemail providing advanced capabilities) to improve communications, boost productivity, and enhance customer

service capabilities across an organization. Cisco Unified Communications solutions also enable users to streamline communication processes through the use of features such as rules-based call routing, simplified contact management, and speech recognition.

- Rich-media conferencing—Enhances the virtual meeting environment with a integrated set of IP-based tools for voice, video, and Web conferencing.
- Video telephony— Enables real-time video communications and collaboration using the same IP network and call processing agent as the Cisco IP Telephony solution. With Cisco Video Telephony, making a video call is now as easy as dialing a phone number.
- Customer contact—Combines strategy and architecture to promote efficient and effective customer communications across a globally-capable network by enabling organizations to draw from a broader range of resources to service customers, including access to an unlimited pool of agents and multiple channels of communication as well as customer self-help tools.
- Third-party applications—Cisco works with leading-edge companies to provide the broadest selection of innovative third-party IP telephony applications and products focused on critical business needs such messaging, customer care, and workforce optimization.

This section describes the deployment of the Cisco IP Telephony solution for a self-managed MPLS MAN network. This topology is provisioned with separate VRFs for voice and data devices. Inter-VPN voice communication occurs over PSTN.

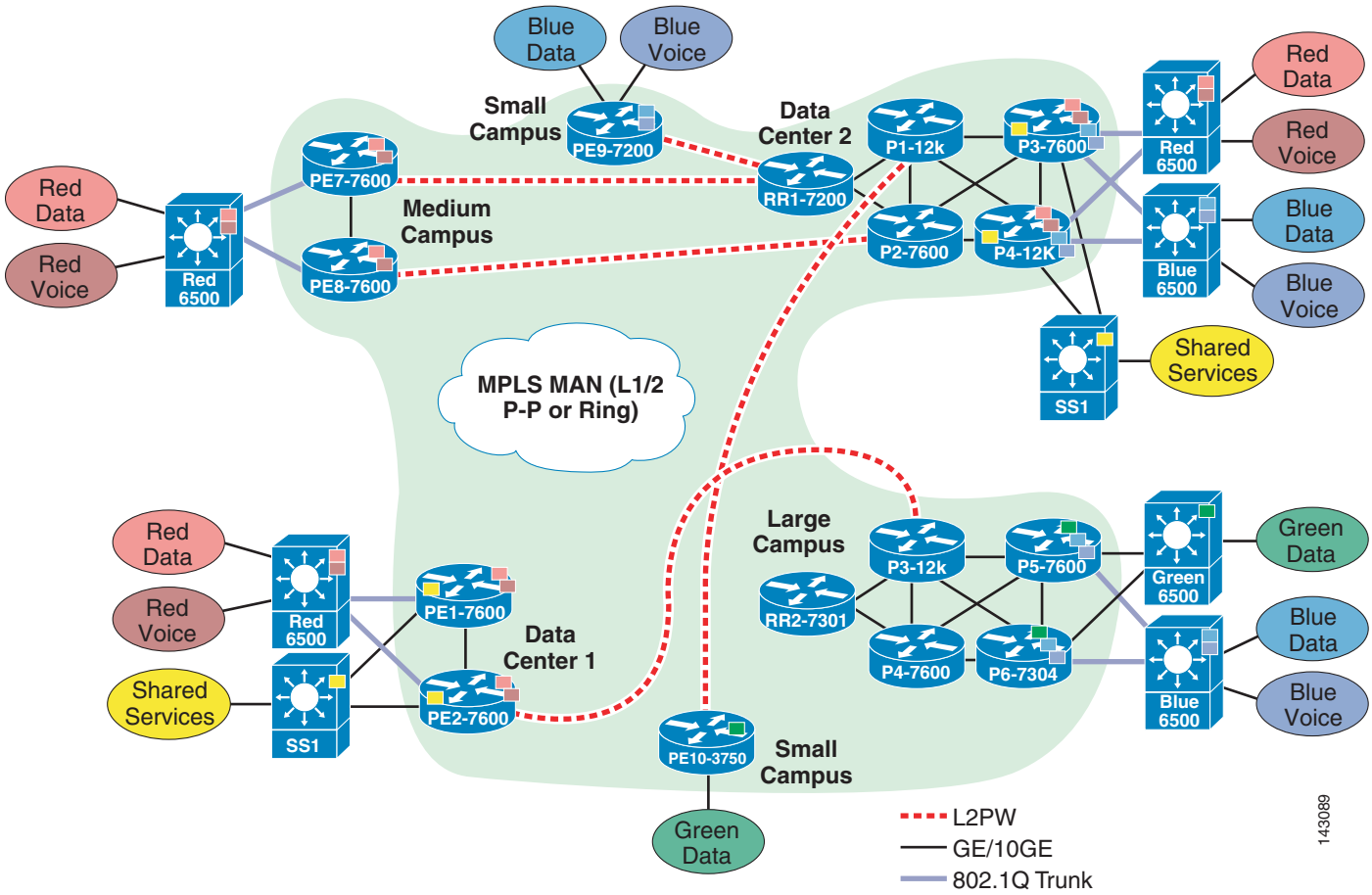
Overview of the Cisco IP Telephony Solution Over the Self-Managed MPLS MAN

The Cisco IP Telephony solution is the leading converged network telephony solution for organizations that want to increase productivity and reduce the costs associated with managing and maintaining separate voice and data networks. The flexibility and sophisticated functionality of the Cisco IP network infrastructure provides the framework that permits rapid deployment of emerging applications such as desktop IP telephony, unified messaging, video telephony, desktop collaboration, enterprise application integration with IP phone displays, and collaborative IP contact centers. These applications enhance productivity and increase enterprise revenues.

The self-managed MPLS MAN enables enterprise customers to begin migration to a more scalable and more efficiently manageable network. It also enables multiple virtual networks on a single network infrastructure (i.e., segmentation).

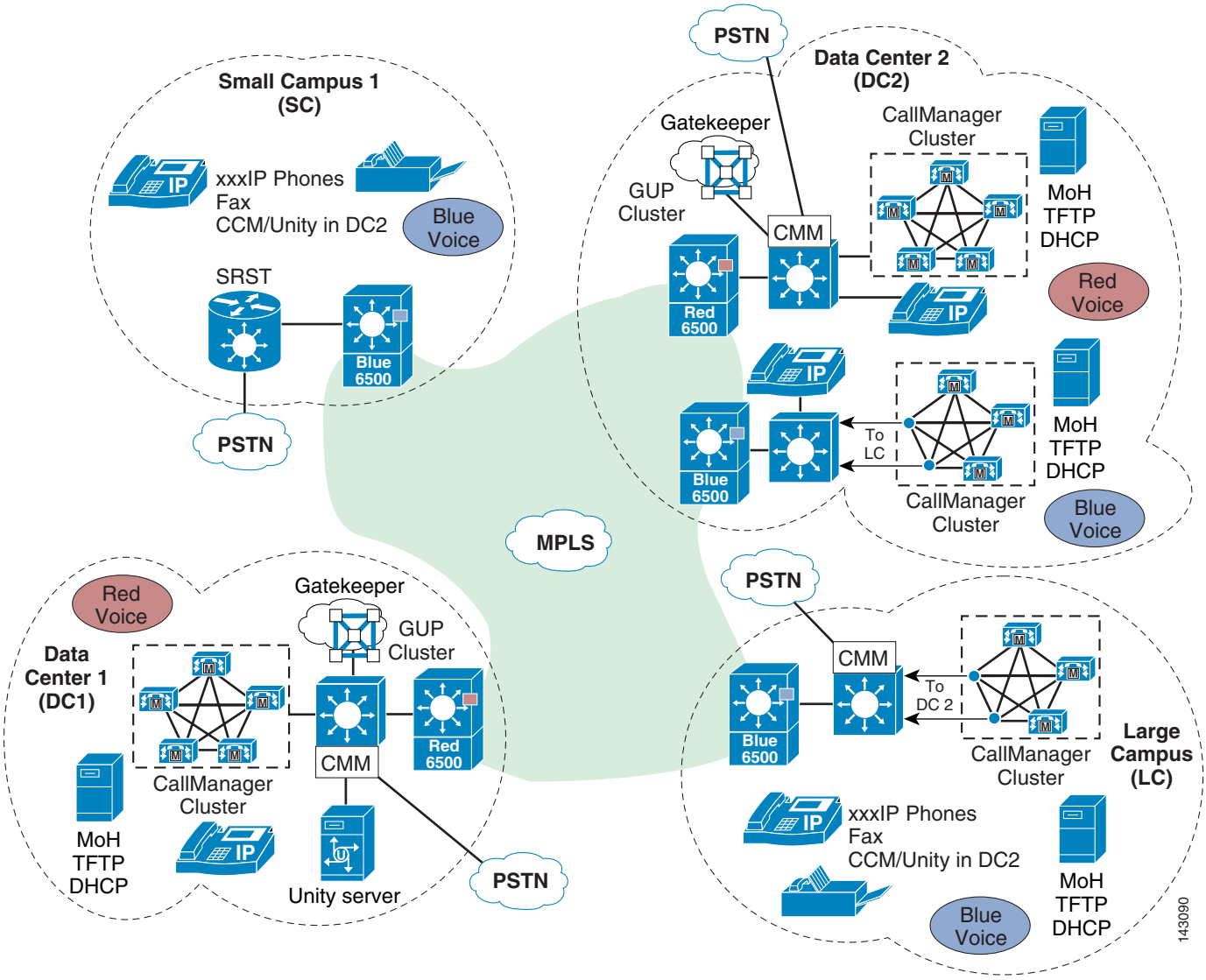
A typical self-managed MPLS MAN allows various organizations and applications to be completely segmented as shown in Figure 6-1.

Figure 6-1 Self-Managed MPLS MAN



This same network with the addition of IP Communications running basic voice servicesis shown in Figure 6-2.

Figure 6-2 Adding IP Communications



In this environment, the network supports multiple organizations segmented across the MPLS MAN segregated by MPLS VPNs. The MPLS VPNs for the voice and data applications are completely segmented with no inter-VPN communications provided.

The foundation architecture of the Cisco IP Telephony solution integrated with the self-managed MPLS MAN includes the following major components:

- Cisco IP network infrastructure
- QoS
- Call processing agent
- Communication endpoints
- Applications

Cisco IP Network Infrastructure

The network infrastructure includes public switched telephone network (PSTN) gateways, analog phone support, and digital signal processor (DSP) farms. The infrastructure can support multiple client types such as hardware phones, software phones, and video devices. The infrastructure also includes the interfaces and features necessary to integrate legacy PBX, voicemail, and directory systems. Typical products used to build the infrastructure include Cisco voice gateways (non-routing, routing, and integrated), Cisco IOS and Cisco Catalyst switches, and Cisco routers.

Quality of Service

Voice, as a class of IP network traffic, has strict requirements concerning packet loss, delay, and delay variation (also known as jitter). To meet these requirements for voice traffic across the MAN, the Cisco IP Telephony solution includes QoS features such as classification, policing, and queuing.

The QoS components of the Cisco IP Telephony solution are provided through the rich IP traffic management, queuing, and policing capabilities of the Cisco IP network infrastructure. Key elements of this infrastructure that enable QoS for IP telephony include:

- Traffic marking
- Enhanced queuing services
- Policing
- Call admission control

Future phases of the design guide will discuss additional QoS tools required to support the WAN and inter-VPN communications.

Call Processing Agent

Cisco CallManager is the core call processing software for the Cisco IP Telephony solution. It builds call processing capabilities on top of the Cisco IP network infrastructure. Cisco CallManager software extends enterprise telephony features and capabilities to packet telephony network devices such as IP phones, media processing devices, Voice over IP (VoIP) gateways, and multimedia applications.

You can deploy the call processing capabilities of Cisco CallManager in the self-managed MPLS MAN according to one of the following models, depending on the size and functional requirements of your enterprise MAN:

- Multi-site MPLS MAN model with distributed call processing—Each site has its own Cisco CallManager cluster for call processing. Communication between sites normally takes place over the IP MPLS MAN, with the PSTN serving as a backup voice path. With this model, you can interconnect any number of sites across the IP MPLS MAN.
- Clustering over the MPLS MAN—You may deploy a single Cisco CallManager cluster across multiple sites that are connected by an MPLS MAN with QoS features enabled. To provide call processing redundancy, you can deploy backup servers either locally at each site or at a remote site across the MPLS MAN. Clustering over the MPLS MAN is well suited as a disaster recovery plan for business continuance sites, perhaps between two data centers.
- Multi-site MPLS MAN model with centralized call processing— The Cisco CallManager cluster resides at a main campus data center. Communication with other offices normally takes place over the MPLS MAN. If either the central site or the MPLS MAN is down, the remote sites can continue to have service through a feature called Survivable Remote Site Telephony (SRST) that runs on Cisco IOS gateways. The remote sites can also place calls over the PSTN if the MPLS MAN is temporarily oversubscribed.

- Hybrid centralized/distributed deployment model across MPLS MAN

The remaining sections of this chapter explain how to apply these deployment models in designing your Cisco IP Telephony network on the self-managed MPLS MAN.

Communication Endpoints

A communication endpoint is a user instrument such as a desk phone or even a software phone application that runs on a PC. In the IP environment, each phone has an Ethernet connection. IP phones have all the functions you expect from a telephone, as well as more advanced features such as the ability to access WWW sites.

In addition to various models of desktop Cisco IP Phones, IP telephony endpoints include the following devices:

- Software-based IP phones

  Cisco IP Communicator and Cisco Softphone are desktop applications that turn your computer into a full-featured IP phone with the added advantages of call tracking, desktop collaboration, and one-click dialing from online directories. Cisco software-based IP phones offer users the great benefit of having a portable office IP phone to use anywhere an Internet connection is available.

- Video telephony endpoints

  Video telephony capability is now fully integrated with Cisco CallManager Release 4.0 and later. In addition, Cisco VT Advantage introduces a Windows-based application and USB camera that can be installed on a Microsoft Windows 2000 or Windows XP personal computer. When the PC is physically connected to the PC port on a Cisco IP Phone 7940, 7960, or 7970, users can make video calls from their IP phones simply by dialing the extension number of another video device on the network.

  Several new third-party video devices are also compatible with the Cisco IP Video Telephony solution.

- Wireless IP Phones

  The Cisco 7920 Wireless IP Phone extends the Cisco family of IP phones from 10/100 Ethernet to 802.11 wireless LAN (WLAN). The Cisco 7920 Wireless IP Phone provides multiple line appearances with functionality similar to existing Cisco 7900 Series IP Phones. In addition, the Cisco 7920 phone provides enhanced WLAN security and Quality of Service (QoS) for operation in 802.11b networks. The Cisco 7920 phone also provides support for XML-based data access and services.

Applications

Voice and video applications build upon the call processing infrastructure to enhance the end-to-end capabilities of the Cisco IP Telephony solution by adding sophisticated telephony and converged network features, such as:

- Unity Messaging

  Cisco Unity delivers powerful unified messaging (email, voice, and fax messages sent to one inbox) and intelligent voice messaging (full-featured voicemail providing advanced functionality) to improve communications, boost productivity, and enhance customer service capabilities across your organization. With Cisco Unity Unified Messaging, you can listen to your email over the phone, check voice messages from the Internet, and (when integrated with a supported third-party fax server) send faxes anywhere.

- Extension mobility

The Cisco CallManager Extension Mobility feature allows users within a Cisco CallManager cluster to configure any Cisco IP Phone 7970, 7960, or 7940 as their own, temporarily, by logging in to that phone. When a user logs in, the phone adopts that user personal phone number(s), speed dials, service links, and other user-specific properties. After logout, the phone reverts to the original user profile. With Cisco CallManager Extension Mobility, several employees can share office space on a rotational basis instead of having a designated office.

- Cisco MeetingPlace

  Cisco MeetingPlace is a complete rich-media conferencing solution that integrates voice, video, and web conferencing capabilities to make remote meetings as natural and effective as face-to-face meetings. In a single step, meeting organizers can schedule voice, video, and web resources through the MeetingPlace web interface, an IP phone, or their Microsoft Outlook or Lotus Notes calendars. Meeting invitees automatically receive notification by email or calendar invitation and can attend rich-media conferences with a single click. With instant messaging applications widely adopted in the workplace, Cisco MeetingPlace also enables users to initiate rich-media conferences easily from common instant messaging clients such as America Online (AOL) Messenger, Lotus Sametime, MSN Messenger, and Yahoo Messenger.

- Web services for Cisco IP Phones

  You can use Cisco IP Phones, such as the Cisco IP Phone 7960 or 7940, to deploy customized client services with which users can interact via the keypad and display. You can create applications for Cisco IP Phone services by using the eXtensible Markup Language (XML) application programming interface (API) and deploy them using the HTTP protocol from standard web servers, such as Microsoft IIS. Some typical services that can be provided through a Cisco IP Phone include a full conferencing interface, the ability to manage data records even if no PC is available, and the ability to display employee alerts, clocks, stock market information, customer contact information, daily schedules, and so forth.

- Cisco IP Contact Center (IPCC) Express

  Cisco IPCC Express is a tightly integrated contact center solution providing three primary functions: interactive voice response (IVR), automatic call distribution (ACD), and computer telephony integration (CTI). The IVR function provides IVR ports to interact with callers by way of either DTMF or speech input. The ACD function provides the ability to intelligently route and queue calls to agents. The CTI function enables call data to be “popped” onto the agent desktop. The IPCC Express software runs on approved Cisco MCS, Hewlett-Packard, or IBM servers and requires interaction with Cisco CallManager.

- Cisco IP Contact Center (IPCC) Enterprise Edition

  Cisco IPCC Enterprise Edition delivers intelligent call routing, network-to-desktop computer telephony integration (CTI), and multi-channel contact management to contact center agents anywhere in the enterprise. The IPCC software profiles each customer using contact-related data such as dialed number and calling line ID, caller-entered digits, data submitted on a Web form, and information obtained from a customer profile database lookup. Simultaneously, the system monitors the resources available in the contact center to meet customer needs, including agent skills and availability, IVR status, queue lengths, and so on. This combination of customer and contact center data is processed through user-defined routing scripts that graphically reflect company business rules, thus enabling Cisco IPCC to route each contact to the optimum resource anywhere in the enterprise.

IP Telephony Deployment Models over the Self-Managed MPLS MAN

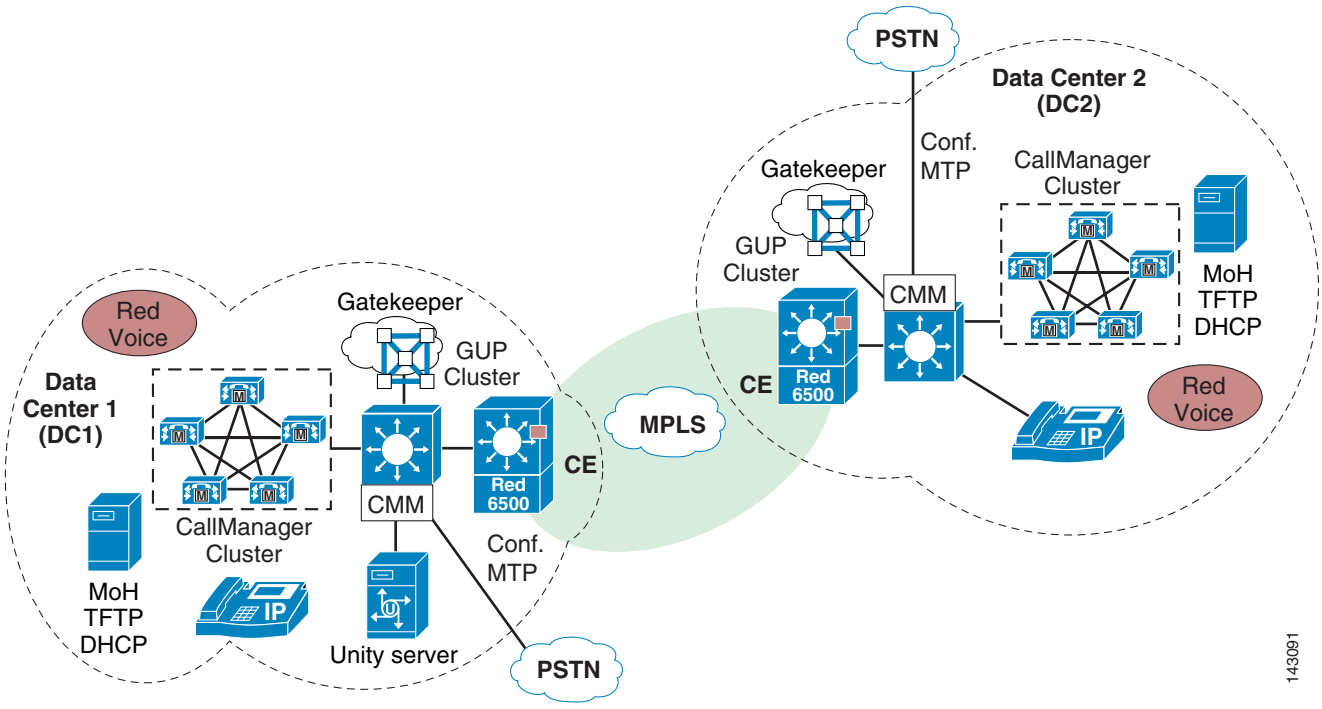
Each Cisco IP Telephony solution is based on one of the following main deployment models described in this chapter:

- Multi-site MPLS MAN model with distributed call processing
- Clustering over the MPLS MAN
- Multi-site MPLS MAN model with centralized call processing
- Hybrid centralized/distributed deployment model across MPLS MAN

Multi-Site MPLS MAN Model with Distributed Call Processing

The multi-site MPLS MAN model with distributed call processing consists of multiple independent sites, each with its own call processing agent connected via the MPLS MAN that carries voice traffic between the distributed sites. Figure 6-3 shows a typical distributed call processing deployment.

Figure 6-3 Typical Distributed Call Processing Deployment



Each site in the distributed call processing model uses its own CallManager cluster for local call processing.

A MPLS MAN interconnects all the distributed call processing sites. Typically, the PSTN is used for off-net calling and serves as a backup connection between the sites in case the MPLS MAN connection fails or does not have any more available bandwidth.

Benefits of the Distributed Call Processing Model

The multi-site MPLS MAN model with distributed call processing provides these benefits:

- Use of the MPLS network for call routing

- Maximum utilization of available bandwidth by allowing voice traffic to share the IP WAN with other types of traffic
- No loss of functionality during a MPLS failure because there is a call processing agent at each site
- Scalability to hundreds of sites

Best Practices for the Distributed Call Processing Model

Follow these guidelines and best practices when implementing the multi-site Distributed Call Processing model:

- Provide a highly available, fault-tolerant infrastructure based on a common infrastructure philosophy. A sound infrastructure is essential for easier migration to IP telephony, integration with applications such as video streaming, and video conferencing.
- Use G.711 codecs for all endpoints. This practice eliminates the consumption of digital signal processor (DSP) resources for transcoding so those resources can be allocated to other functions such as conferencing and Media Termination Points (MTPs).
- Use Media Gateway Control Protocol (MGCP) gateways for the PSTN if you do *not* require H.323 functionality. This practice simplifies the dial plan configuration. H.323 might be required to support specific functionality not offered with MGCP, such as support for Signaling System 7 (SS7) or Non-Facility Associated Signaling (NFAS).
- Implement the recommended network infrastructure for high availability, connectivity options for phones (in-line power), Quality of Service (QoS) mechanisms, and security. See the Enterprise IP Telephony SRND at the following URL:

  http://www.cisco.com/en/US/products/sw/voicesw/ps556/products_implementation_design_guide_book09186a008044714e.html
- Follow the provisioning recommendations listed in the Enterprise IP telephony SRND chapter on Call Processing:
  http://www.cisco.com/en/US/products/sw/voicesw/ps556/products_implementation_design_guide_book09186a008044714e.html

Gatekeeper or Session Initiation Protocol (SIP) proxy servers are among the key elements in the multi-site MPLS MAN model with distributed call processing. They each provide dial plan resolution, with the gatekeeper also providing call admission control, although bandwidth is not normally a limitation for transporting voice in a MAN design. A gatekeeper is an H.323 device that provides call admission control and E.164 dial plan resolution.

In multi-site deployments where a Cisco CallManager cluster is present at each site and the sites are linked through the MPLS MAN, a gatekeeper can provide call admission control between the sites, with each site being placed in a different gatekeeper zone.

Bandwidth would be a rare instance because bandwidth is not normally an issue when interconnecting telephony sites across the MPLS MAN; however, there is such a requirement for Call Admission Control, when all the available bandwidth that is provisioned for voice between particular sites has been utilized, you can provide automatic failover to the PSTN using the route list and route group construct for the route patterns that connect each cluster to the gatekeeper. For more detail on automatic failover see the Enterprise IP Telephony SRND at the following URL:
http://www.cisco.com/en/US/products/sw/voicesw/ps556/products_implementation_design_guide_book09186a008044714e.html.

The following best practices apply to the use of a gatekeeper:

- Use a Cisco IOS gatekeeper to provide call admission control into and out of each site.

- To provide high availability of the gatekeeper, use Hot Standby Router Protocol (HSRP) gatekeeper pairs, gatekeeper clustering, and alternate gatekeeper support. In addition, use multiple gatekeepers to provide redundancy within the network.
- Size the platforms appropriately to ensure that performance and capacity requirements can be met.
- Because this is a MAN deployment and bandwidth is plentiful, use the single G.711 codec on the MPLS MAN.
- Gatekeeper networks can scale to hundreds of sites and the design is limited only by the MAN topology.

For more information on the various functions performed by gatekeepers, see the following sections such as scalability, redundancy, and dial plan resolution in the Enterprise IP Telephony SRND at the following URL: http://www.cisco.com/en/US/products/sw/voicesw/ps556/products_implementation_design_guide_book09186a008044714e.html.

SIP devices provide resolution of E.164 numbers as well as SIP uniform resource identifiers (URIs) to enable endpoints to place calls to each other. Cisco CallManager supports the use of E.164 numbers only.

The following best practices apply to the use of SIP proxies:

- Provide adequate redundancy for the SIP proxies.
- Ensure that the SIP proxies have the capacity for the call rate and number of calls required in the network.

Note Planning for call admission control is outside the scope of this document.

Clustering over the MPLS MAN

You may deploy a single Cisco CallManager cluster across multiple sites, such as two data centers, that are connected by the MPLS MAN. This section provides a brief overview of clustering over the MAN.

For further information, see the Enterprise IP Telephony SRND at the following URL: http://www.cisco.com/en/US/products/sw/voicesw/ps556/products_implementation_design_guide_book09186a008044714e.html.

Clustering over the WAN can support two types of deployments:

- Local Failover Deployment Model

  Local failover requires that you place the Cisco CallManager subscriber and backup servers at the same site with no WAN between them. This deployment model is ideal for two to four sites with Cisco CallManager

- Remote Failover Deployment Model

  Remote failover allows you to deploy the backup servers across the MPLS MAN. Using this deployment model, you may have up to eight sites with Cisco CallManager subscribers being backed up by Cisco CallManager subscribers at another site.

You can also use a combination of the two deployment models to satisfy specific site requirements. For example, two main sites may each have primary and backup subscribers, with another two sites containing only a primary server each and using either shared backups or dedicated backups at the two main sites.

The key advantages of clustering over the WAN are:

- Single point of administration for users for all sites within the cluster

- Feature transparency
- Shared line appearances
- Extension mobility within the cluster
- Unified dial plan

These features make this solution ideal as a disaster recovery plan for business continuance sites or as a single solution for up to eight small or medium sites.

MPLS MAN Considerations

For clustering over the MPLS MAN to be successful, you must carefully plan, design, and implement various characteristics of the MAN itself. The Intra-Cluster Communication Signaling (ICCS) between Cisco CallManager servers consists of many traffic types. The ICCS traffic types are classified as either Priority or Best Effort. Priority ICCS traffic is marked with IP Precedence 3 (DSCP 26 or PHB AF31 in Cisco CallManager releases before 4.0 and DSCP 24 or PHB CS3 for Release 4.0 and later). Best Effort ICCS traffic is marked with IP Precedence 0 (DSCP 0 or PHB BE). The various types of ICCS traffic are described in the subsequent Intra-Cluster Communications section, which also provides further guidelines for provisioning.

Delay

Because delay is minimal across the MPLS MAN, details are not presented in this section. However, it is still important to understand the delay requirements for designing a cluster across the MAN/WAN. For details, see the Enterprise IP Telephony SRND at the following URL: http://www.cisco.com/en/US/products/sw/voicesw/ps556/products_implementation_design_guide_book09186a008044714e.html.

Jitter

Jitter is the varying delay that packets incur through the network because of processing, queue, buffer, congestion, or path variation delay. Jitter for the IP Precedence 3 ICCS traffic must be minimized using QoS features.

Packet Loss and Errors

The network should be engineered for zero percent packet loss and errors for all ICCS, especially the priority ICCS traffic, because packet loss and errors have adverse effects on the real-time call processing within the cluster.

Bandwidth

Bandwidth is not normally a cause of concern in the MPLS MAN environment. Typically, you provision the correct amount of bandwidth between each server for the expected call volume, type of devices, and number of devices. This bandwidth is in addition to any other bandwidth for other applications sharing the network, including voice and video traffic between the sites. The bandwidth provisioned must have QoS enabled to provide the prioritization and scheduling for the different classes of traffic. The general rule of thumb for bandwidth is to over-provision and under-subscribe.

Quality of Service

The network infrastructure relies on QoS engineering to provide consistent and predictable end-to-end levels of service for traffic. Neither QoS nor bandwidth alone is the solution; rather, QoS-enabled bandwidth must be engineered into the network infrastructure.

Intra-Cluster Communications

In general, intra-cluster communications means all traffic between servers. There is also a real-time protocol called Intra-Cluster Communication Signaling (ICCS) that provides the communications with the Cisco CallManager Service process that is at the heart of the call processing in each server or node within the cluster.

The intra-cluster traffic between the servers consists of:

- Database traffic from the SQL database that provides the main configuration information. The SQL database is replicated from the publisher server to all other servers in the cluster using Best Effort. The SQL traffic may be re-prioritized in line with Cisco QoS recommendations to a higher priority data service (for example, IP Precedence 1 if required by the particular business needs). An example of this is extensive use of Extension Mobility that relies on SQL database configuration.
- Directory traffic from the Lightweight Directory Access Protocol (LDAP) directory provides user and application authentication and some additional specific user or application configuration information. LDAP traffic is sent Best Effort by default.
- ICCS real-time traffic, which consists of signaling, call admission control, and other information regarding calls as they are initiated and completed. ICCS uses a TCP connection between all servers that have the Cisco CallManager Service enabled. The connections are a full mesh between these servers. Because only eight servers may have the Cisco CallManager Service enabled in a cluster, there may be up to seven connections on each server. This traffic is priority ICCS traffic and is marked dependant on release and service parameter configuration.
- CTI Manager real-time traffic is used for CTI devices involved in calls or for controlling or monitoring other third-party devices on the Cisco CallManager servers. This traffic is marked as priority ICCS traffic and exists between the Cisco CallManager server with the CTI Manager and the Cisco CallManager server with the CTI device.

Failover between Subscriber Servers

With Cisco CallManager Release 3.1 and 3.2, failover behavior is dependant on the reachability of the publisher and the delay between the subscriber and the publisher. If the publisher is reachable, the subscriber requests the relevant device configuration records directly from the publisher during device registration. The round-trip delay and the available bandwidth for the SQL database traffic affects the speed of registrations. The effect of this is that failover for devices at remote locations to the publisher may experience delays of approximately 20 minutes before all devices on a full server complete the failover process. If the publisher is unreachable during failover, the subscriber uses its own most recent copy of the database for the configuration information. Because there is no incurred delay for the subscriber to access its own database, the failover time in this case is approximately five minutes for a full server.

With Cisco CallManager Release 3.3 and higher, the impact of delay to the publisher is minimized during the failover period because the configuration information is cached during initialization or boot-up time. The effect is that the Cisco CallManagers might take longer to start up initially; however any subsequent failover and failback is not affected by the delay in accessing the publisher database.

Cisco CallManager Publisher

The publisher replicates a read-only copy of the master database to all other servers in the cluster. If changes are made in the publisher master database during a period when another server in the cluster is unreachable, the publisher replicates the updated database when communications are re-established.

During any period when the publisher is unreachable or offline, no changes can be made to the configuration database. All subscriber databases are read-only and may not be modified. Most normal operations of the cluster are not affected during this period, including:

- Call processing
- Failover
- Installation registration of previously configured devices

There are some features and functions that require access to the master database on the publisher because they make modifications to records and therefore need write access. The publisher is the only server in a Cisco CallManager cluster that has a read and write configuration database. The main features and functions that require access to the publisher for write access include:

- Configuration additions, changes, and deletions
- Extension Mobility
- User speed dials
- Cisco CallManager User page options requiring the database
- Cisco CallManager software upgrades
- Call Forward All changes
- Message Waiting Indicator (MWI) state

Other services or applications might also be affected and their ability to function without the publisher should be verified when deployed.

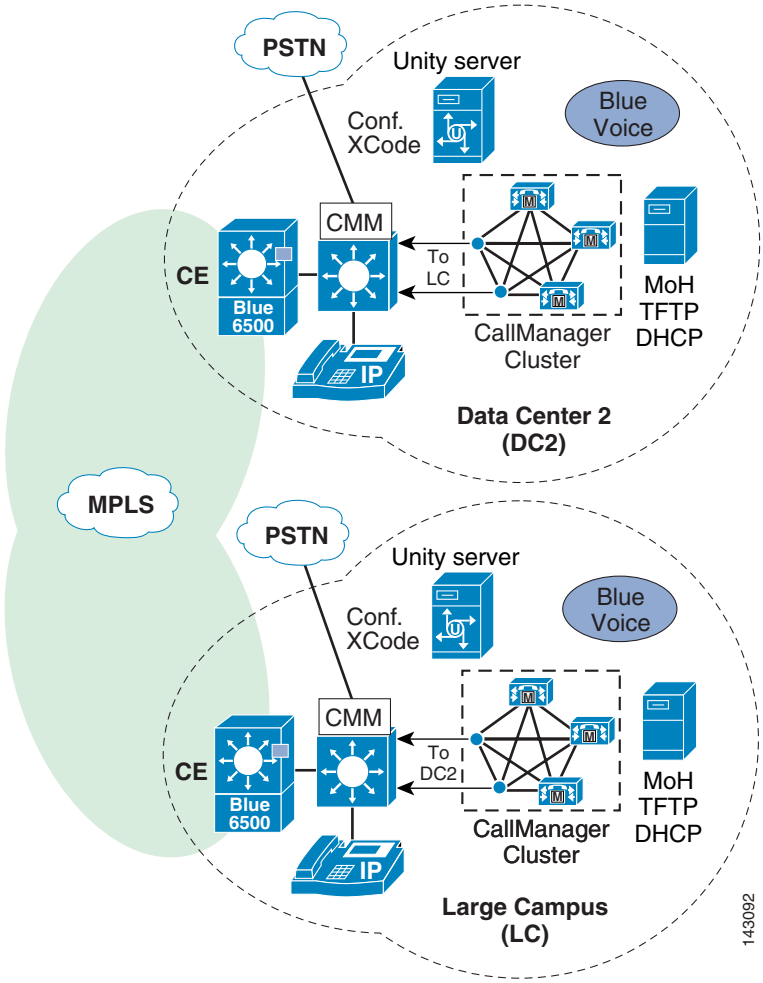
Call Detail Records (CDR)

Call detail records, when enabled, are collected by each subscriber and uploaded to the publisher periodically. During a period when the publisher is unreachable, the CDRs are stored on the subscriber local hard disk. When connectivity is re-established to the publisher, all outstanding CDRs are uploaded to the publisher.

Local Failover Deployment Model

The local failover deployment model provides the most resilience for clustering the sites in this model contains at least one primary Cisco CallManager subscriber and one backup subscriber. This configuration can support up to four sites. The maximum number devices is dependant on the quantity and type of servers deployed. The maximum IP phones for all sites is 30,000 (see Figure 6-4).

Figure 6-4 Local Failover Deployment Model

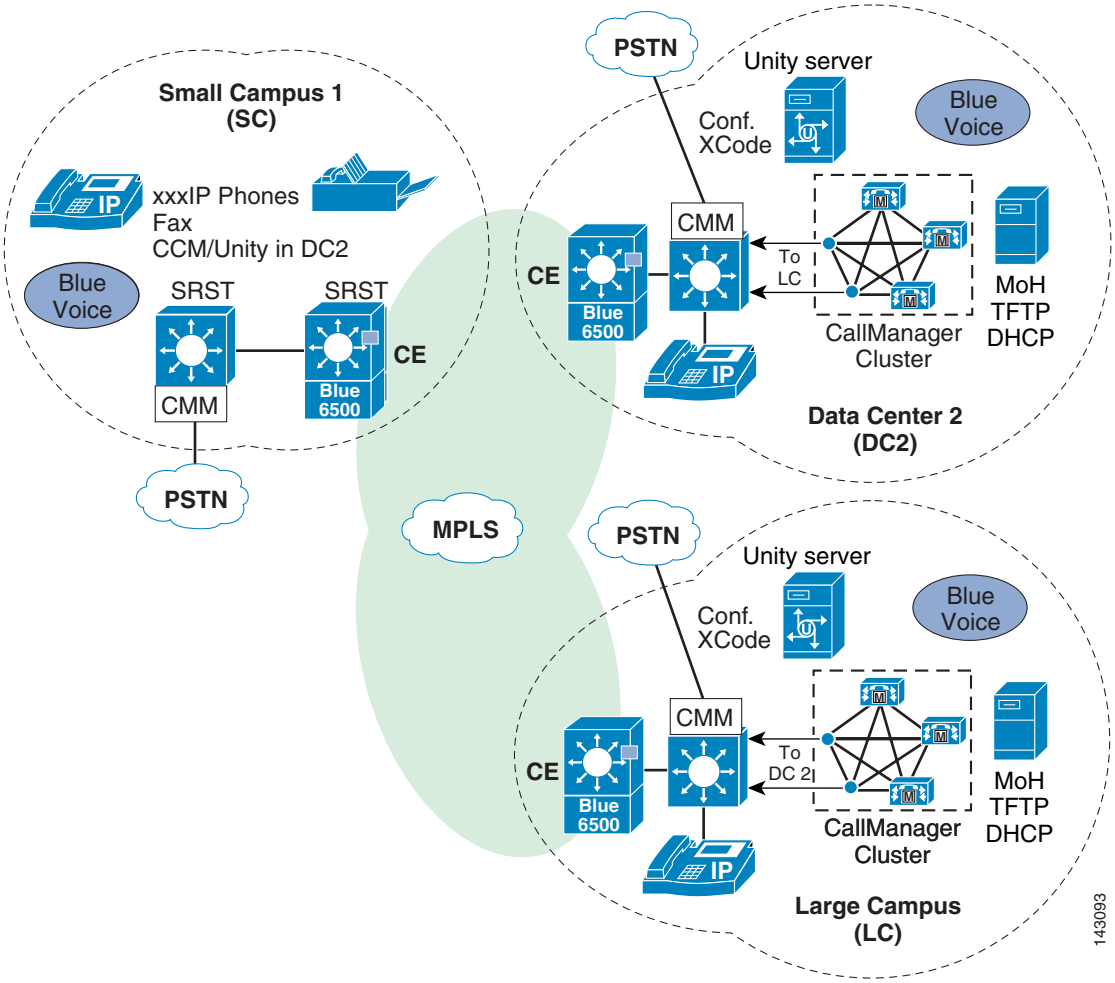


Observe the following guidelines when implementing the local failover model:

- Configure each site to contain at least one primary Cisco CallManager subscriber and one backup subscriber.
- Configure Cisco CallManager *groups* and *device pools* to allow devices within the site to register with only the servers at that site under all conditions.
- Cisco highly recommends that you replicate key services (TFTP, DNS, DHCP, LDAP, and IP Phone services), all media resources (conference bridges and music on hold), and gateways at each site to provide the highest level of resiliency. You can also extend this practice to include a voicemail system at each site.
- Under a failure condition, sites without access to the publisher database lose some functionality:
  - System administration at the local site is not able to add, modify, or delete any part of the configuration.
  - Extension mobility users are not able to log in or log out of the IP phones.
  - Changes to Call Forward All are not allowed.

- Under MPLS MAN failure conditions, calls made to phone numbers that are not currently communicating with the subscriber placing the call result in either a fast-busy tone or a call forward (possibly to voicemail, depending on the location of the phone number to which they are being forwarded). During this condition, users should manually dial those numbers via the PSTN.
- Every 10,000 busy hour call attempts (BHCA) between sites that are clustered over the WAN requires 900 kbps of bandwidth for ICCS. This is a minimum bandwidth requirement and bandwidth is allocated in multiples of 900 kbps. The ICCS traffic types are classified as either Priority or Best Effort. Priority ICCS traffic is marked with IP Precedence 3 (DSCP 26 or PHB AF31 in Cisco CallManager releases before 4.0 and DSCP 24 or PHB CS3 for Release 4.0 and later). Best Effort ICCS traffic is marked with IP Precedence 0 (DSCP 0 or PHB BE).
- The local failover model requires Cisco CallManager Release 3.1 or later.
- Because the MAN is based on MPLS, any site connected to the MAN that does not have a local CallManager subscriber for call processing may have those telephony devices register with any CallManager subscriber at either of the main CallManager cluster sites (see Figure 6-5).

Figure 6-5 Remote Subscription to CallManager



- During a software upgrade, all servers in the cluster should be upgraded during the same maintenance period using the standard upgrade procedures outlined in the software release notes.

Cisco CallManager Provisioning for Local Failover

Provisioning of the Cisco CallManager cluster for the local failover model should follow the design guidelines for capacities outlined in the chapter on Call Processing in the Enterprise IP Telephony SRND at the following URL: http://www.cisco.com/en/US/products/sw/voicesw/ps556/products_implementation_design_guide_book09186a008044714e.html.

If voice or video calls are allowed across the MPLS MAN between the sites, then you must configure Cisco CallManager *locations*, in addition to the default location for the other sites, to provide call admission control between the sites. Even though the bandwidth is more than likely over-provisioned across the MPLS MAN for the number of devices, it is still best practice to configure call admission control based on locations. If the locations-based call admission control rejects a call, automatic failover to the PSTN can be provided by the automated alternate routing (AAR) feature.

To improve redundancy and upgrade times, Cisco recommends that you enable the Cisco TFTP service on at least one of the Cisco CallManager servers at each location. You can run the TFTP service on either a publisher or a subscriber server, depending on the site and the available capacity of the server. The TFTP server option must be correctly set on the DHCP servers for each site. If DHCP is not in use or the TFTP server is manually configured, you should configure the correct address for the site.

Other services which may affect normal operation of Cisco CallManager during MPLS MAN outages should also be replicated at all sites to ensure uninterrupted service. These services include DHCP servers, DNS servers, corporate directories, and IP phone services. On each DHCP server, set the DNS server address correctly for each location.

Gateways for Local Failover

Normally, gateways should be provided at all sites for access to the PSTN. The device pools should be configured to register the gateways with the Cisco CallManager servers at the same site. Partitions and calling search spaces should also be configured to select the local gateways at the site as the first choice for PSTN access and the other site gateways as a second choice for overflow. Take special care to ensure emergency service access at each site.

You can centralize access to the PSTN gateways if access is not required during a MAN failure. For E911 requirements, additional gateways might be needed at each site.

Voicemail for Local Failover

Cisco Unity or other voicemail systems can be deployed at all sites and integrated into the Cisco CallManager cluster. This configuration provides voicemail access even during a MAN failure and without using the PSTN.

Using Voice Mail Profiles, you can allocate the correct voicemail system for the site to the IP phones in the same location. You can configure a maximum of four voicemail systems per cluster that use the SMDI protocol, which are attached directly to the COM port on a subscriber and that use the Cisco Messaging Interface (CMI).

Music on Hold and Media Resources for Local Failover

Music on hold (MoH) servers and other media resources such as conference bridges should be provisioned at each site with sufficient capacity for the type and number of users. Through the use of media resource groups (MRGs) and media resource group lists (MRGLs), media resources are provided by the on-site resource and are available during a MAN failure.

The remote failover deployment model provides flexibility for the placement of backup servers. Each of the sites contains at least one primary Cisco CallManager subscriber and may or may not have a backup subscriber. This model allows for a deployment of up to eight sites with IP phones and other devices normally registered to a local subscriber when using 1:1 redundancy and the 50/50 load balancing option described in the Enterprise IP Telephone SRND at the following URL: http://www.cisco.com/en/US/products/sw/voicesw/ps556/products_implementation_design_guide_book09186a008044714e.html.

Backup subscribers are located across the MPLS MAN at one or more of the other sites.

When implementing the remote failover model, observe all guidelines for the local failover model with the following modifications:

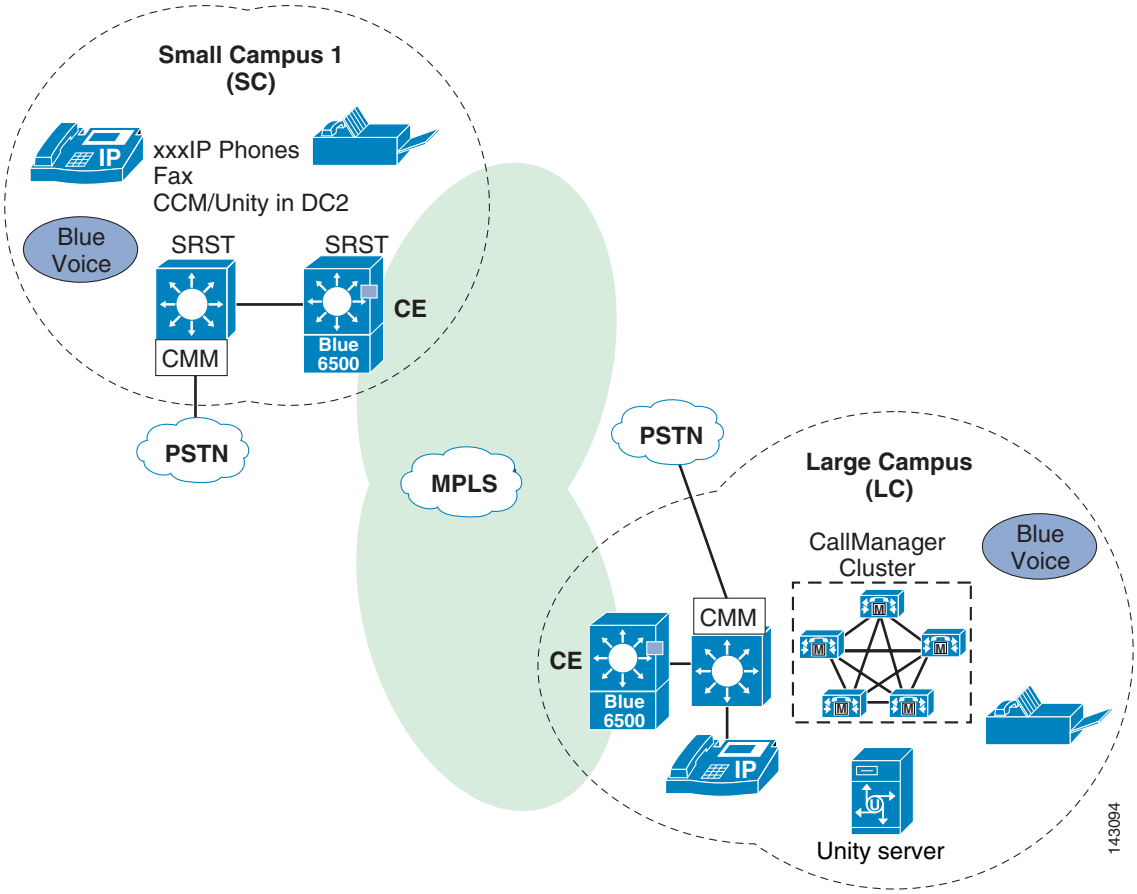
- Configure each site to contain at least one primary Cisco CallManager subscriber and an optional backup subscriber as desired.
- You may configure Cisco CallManager *groups* and *device pools* to allow devices to register with servers over the MPLS MAN.

Multi-Site MPLS MAN Model with Centralized Call Processing

The multi-site MPLS MAN model with centralized call processing consists of a single call processing agent that provides services for many sites and uses the MPLS MAN to transport IP telephony traffic between the sites. The MPLS MAN also carries call control signaling between the central site and the remote sites.

Figure 6-6 illustrates a typical centralized call processing deployment, with a Cisco CallManager cluster as the call processing agent at a data center central site and an IP MPLS MAN with QoS enabled to connect one or multiple additional sites.

Figure 6-6 Multi-site MPLS MAN Model with Centralized Call Processing



The campus sites remote from CallManager rely on the centralized Cisco CallManager cluster to handle their call processing. Applications such as voicemail are typically centralized as well to reduce the overall costs of administration and maintenance.

Note In each solution for the centralized call processing model presented in this document, the various sites connect to a MPLS MAN with QoS configured.

Routers and switches that reside at the edges require QoS mechanisms, such as priority queuing and policing to protect the voice traffic from the data traffic across the MPLS MAN. In addition, a call admission control scheme is needed to avoid oversubscribing the links with voice traffic and deteriorating the quality of established calls.

For centralized call processing deployments, the *locations* construct within Cisco CallManager provides call admission control. See the section on Cisco CallManager Locations; for more information on locations see the Enterprise IP Telephony SRND at the following URL: http://www.cisco.com/en/US/products/sw/voicesw/ps556/products_implementation_design_guide_book09186a008044714e.html.

A variety of Cisco gateways can provide the remote sites with PSTN access. If all the available bandwidth allocated for voice on the MPLS MAN has been consumed, the system uses AAR to re-route calls between sites across the PSTN.

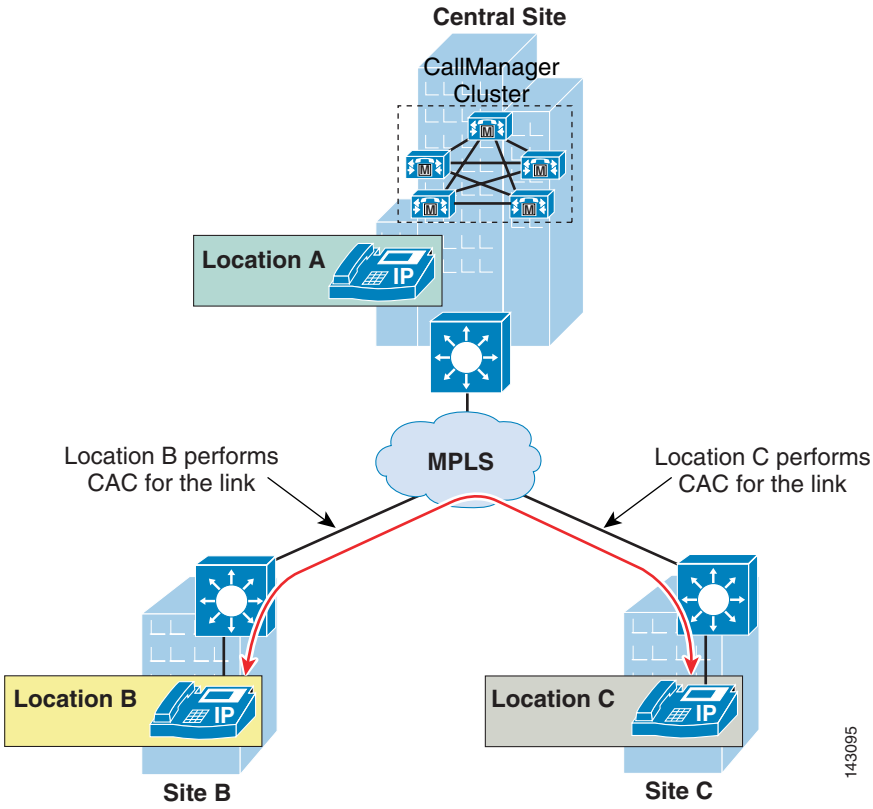
The Survivable Remote Site Telephony (SRST) feature, available on Cisco IOS gateways, provides call processing at the remote offices from CallManager in the event of a MPLS MAN failure. Users at each remote site can dial the PSTN access code and place their calls through the PSTN.

Best Practices for the Centralized Call Processing Model

Follow these guidelines and best practices when implementing the multi-site MPLS MAN model with centralized call processing:

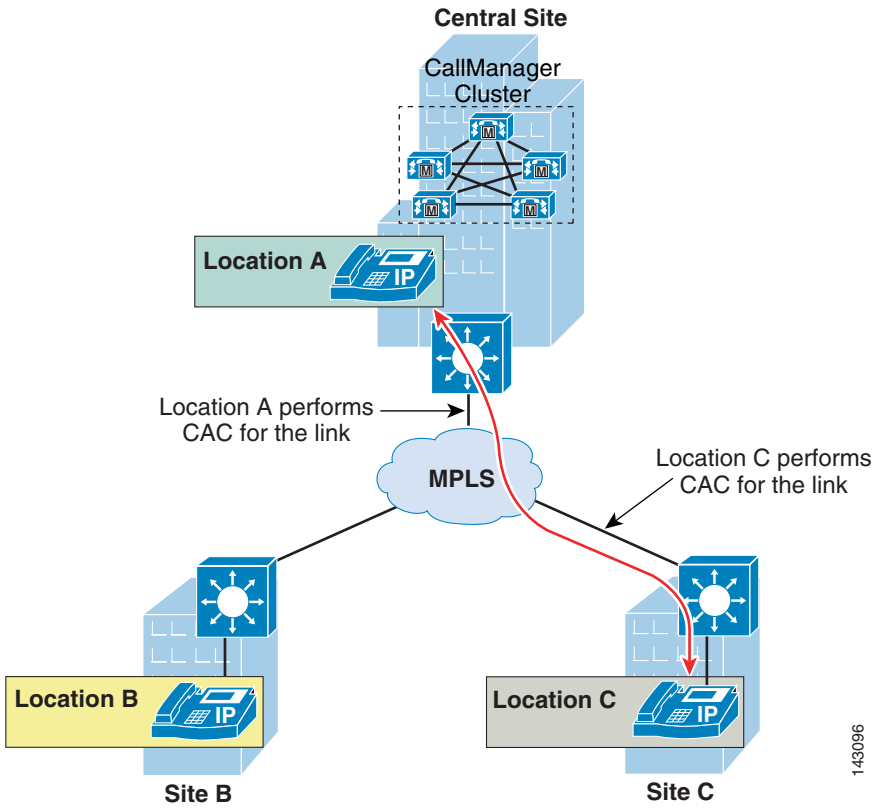
- In single-cluster centralized call processing deployments, the call admission control (CAC) function is performed by the *locations* construct within Cisco CallManager. Although in a MPLS MAN environment there would most likely not be bandwidth issues requiring CAC if there is such a requirement follow these best practices.
- With an MPLS network, all sites connected across the MAN are deemed to be adjacent at Layer 3, thus they do not have to rely on the central site for connectivity.

Figure 6-7 CAC for Calls Between Remote Sites



- Also, in an MPLS MAN, the link connecting the central Call Processing site does not aggregate every remote site link. Place all the central site devices in their own call admission control location (that is, not in the <None> location); this configuration requires that call admission control be performed on the central site link independently of the remote site links.

Figure 6-8 CAC for Calls Between Central Site and Remote Sites



- When all the available bandwidth reserved for voice for a particular site has been utilized, you can provide automatic failover to the PSTN using the automated alternate routing (AAR) feature within Cisco CallManager. For more information on AAR, see the section on Automated Alternate Routing in the Enterprise IP Telephony SRND at the following URL: http://www.cisco.com/en/US/products/sw/voicesw/ps556/products_implementation_design_guide_book09186a008044714e.html.
- Use the locations mechanism in Cisco CallManager to provide call admission control into and out of sites that do not have CallManager servers.
- The locations mechanism works across multiple servers in Cisco CallManager Release 3.1 and later.
- This configuration can support a maximum of 30,000 IP phones when Cisco CallManager runs on the largest supported server.
- The number of IP phones and line appearances supported in SRST mode at each remote site depends on the router or Catalyst gateway module used, the amount of memory installed, and the Cisco IOS release. (For the latest SRST platform and code specifications, see the SRST documentation at http://www.cisco.com.) Generally speaking, however, the choice of whether to adopt a centralized call processing or distributed call processing approach for a given site depends on a number of factors such as:
  - Criticality of the voice network
  - Feature set needs
  - Scalability
  - Ease of management

- Cost

If a distributed call processing model is deemed more suitable for customer business needs, you would include in the design the installation of a local Cisco CallManager cluster at each location or design a CallManager cluster across multiple locations as described above.

Survivable Remote Site Telephony

When deploying IP telephony across a self-managed MPLS MAN with the centralized call processing model, the MAN is highly available based on the robustness to the MPLS design.

However, should there be a catastrophic outage of the network isolating the remote locations from the centralized CallManager cluster, SRST provides high availability for voice services only by providing a subset of the call processing capabilities within the remote office location and enhancing the IP phones with the ability to "re-home" to the call processing functions in the local network infrastructure device if a the MPLS network failure is detected.

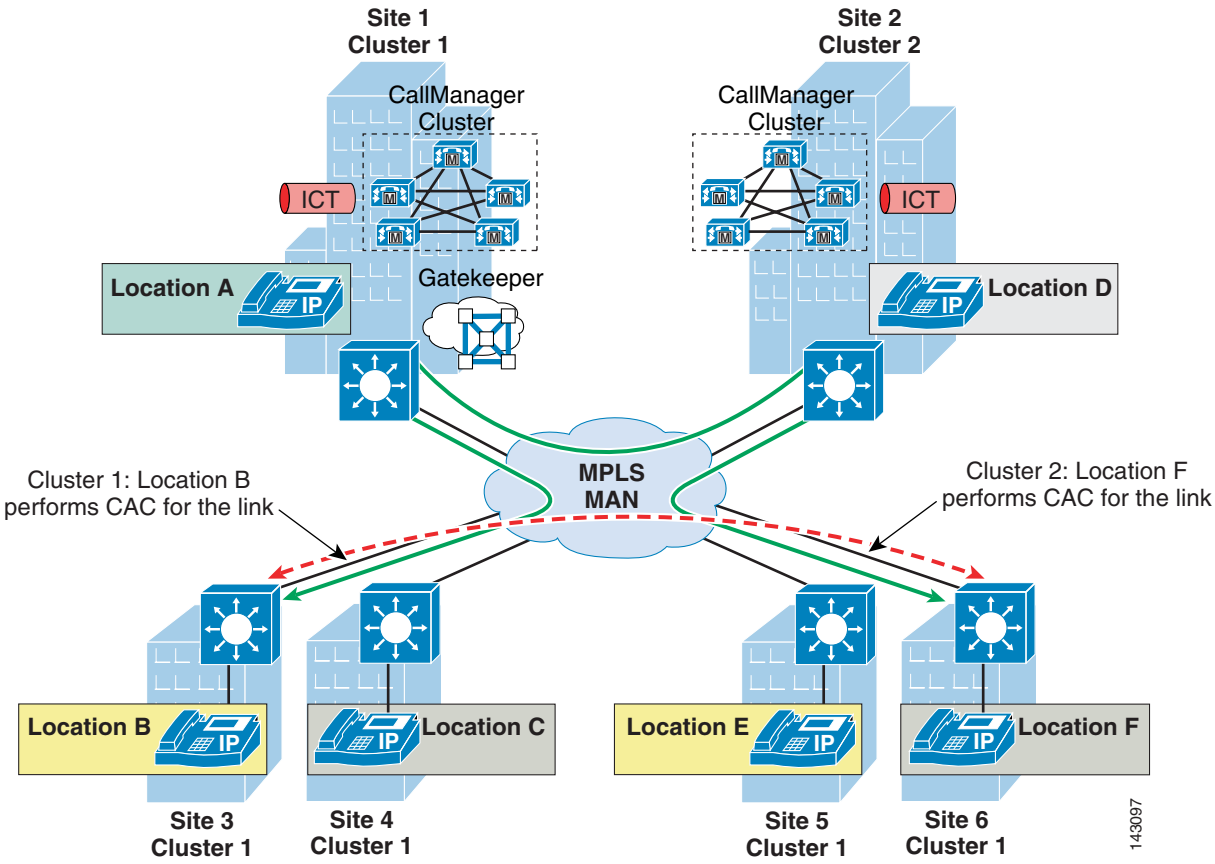
For more detail of SRST functionality in a IP Communications centralized call processing environment, see the following URL: http://www.cisco.com/en/US/products/sw/voicesw/ps556/products_implementation_design_guide_book09186a008044714e.html.

Hybrid Centralized/Distributed Deployment Model across MPLS MAN

For multi-site deployments that combine both centralized and distributed call processing deployment models, a MPLS MAN presents a new situation for inter-cluster calls.

When calls occur between two sites belonging to different clusters, the audio path is established between the two sites directly with no media transiting through each cluster central site. Therefore call admission control is required only on the link at the two remote sites (see Figure 6-9) .

Figure 6-9 CAC for Distributed Deployment Model



As in the purely centralized deployments, devices that terminate media at each site (including the central sites for each cluster) must be placed in an appropriately configured location.

Note that the inter-cluster trunks are purely signaling devices and there is no media transiting through them. Therefore all inter-cluster trunks must be left in location <None>.

In these deployments, a gatekeeper can be used for dial plan resolution between clusters, but a gatekeeper is not recommended for call admission control.

Although bandwidth is normally not an issue to carry voice in a MPLS MAN environment, if the available bandwidth provisioned for voice for a particular site has been used, you can provide automatic failover to the PSTN by using a combination of the following two methods:

- The route list and route group construct for calls across multiple Cisco CallManager clusters
- The automated alternate routing (AAR) feature for calls within a Cisco CallManager more information on AAR. See the “Automated Alternate Routing” section in the Enterprise IP telephony SRND at the following URL: http://www.cisco.com/en/US/products/sw/voicesw/ps556/products_implementation_design_guide_book09186a008044714e.html.

Network Infrastructure

This section describes the requirements of the network infrastructure needed to build an IP telephony system in an enterprise environment. Figure 6-10 shows the roles of the various devices that form the network infrastructure.

Figure 6-10 Network Infrastructure

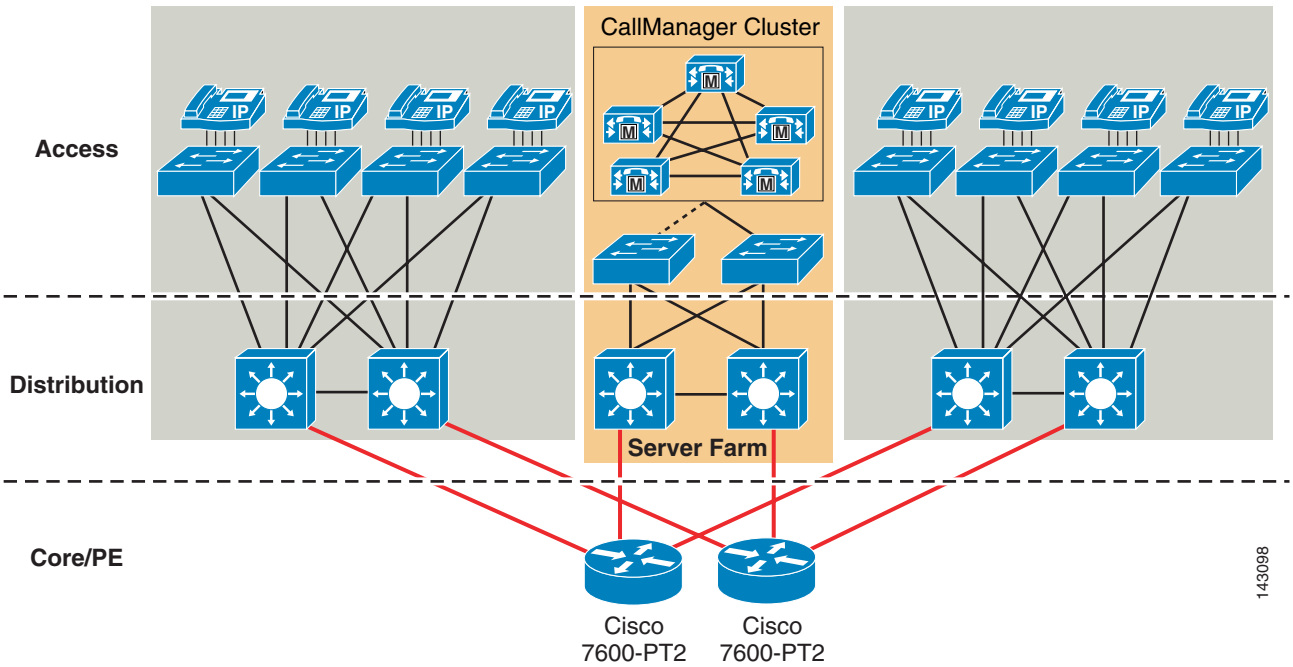


Table 6-1 summarizes the features required to support each of these roles.

Table 6-1 Required Features

Infrastructure Role	Required Feature
Campus access switch	In-line power
	Multiple queue support
	802.1p and 801.2Q
	Fast link convergence
Campus distribution	Multiple queue support
	VRF lite
	Traffic classification
	Traffic re-classification
PE	Multiple queue support
	VRF
	Traffic classification
	Traffic re-classification

Campus Access Layer

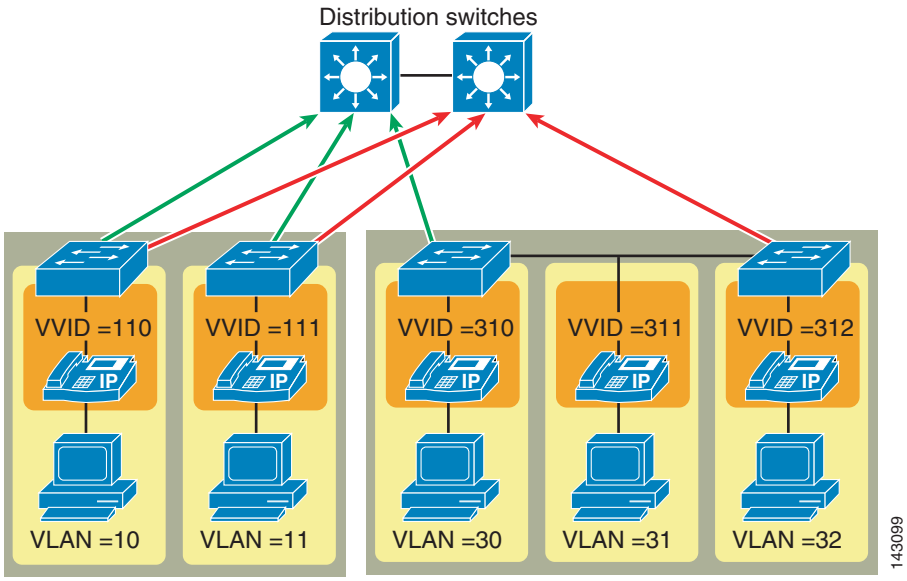
This section focuses on the campus access layer. The remaining blocks and component of self-managed MPLS infrastructure are addressed in the Next Generation MPLS Architecture and QoS sections of this design guide in addition to the appropriate Enterprise IP Telephony SRND at the following URL: http://www.cisco.com/en/US/products/sw/voicesw/ps556/products_implementation_design_guide_book09186a008044714e.html.

The access layer of the Campus LAN includes the portion of the network from the desktop port(s) to the wiring closet switch.

It is currently required to design the access layer using traditional design methodology to the distribution layer switches because many of the features required for voice applications, for example SRST, Gateways, DSP Resources, and so on, currently are not VRF-aware.

Proper access layer design starts with assigning a single IP subnet per virtual LAN (VLAN). Typically, a VLAN should not span multiple wiring closet switches; that is, a VLAN should have presence in one and only one access layer switch (see Figure 6-11). This practice eliminates topological loops at Layer 2, thus avoiding temporary flow interruptions because of Spanning Tree convergence. However with the introduction of standards-based IEEE 802.1w Rapid Spanning Tree Protocol (RSTP) and 802.1s Multiple Instance Spanning Tree Protocol (MISTP), Spanning Tree can converge at much higher rates.

Figure 6-11 Campus Access Layer



In situations where RSTP and/or MISTP can and have been configured on the access layer switch, there is no need for concern about topological loops. More importantly, confining a VLAN to a single access layer switch also serves to limit the size of the broadcast domain. There is the potential for large numbers of devices within a single VLAN or broadcast domain to generate large amounts of broadcast traffic periodically, which can be problematic. A good rule of thumb is to limit the number of devices per VLAN to about 512, which is equivalent to two Class C subnets (that is, a 23-bit subnet masked Class C address).

When you deploy voice, Cisco recommends that you enable two VLANs at the access layer: a native VLAN for data traffic (VLANs 10, 11, 30, 31, and 32 in Figure 6-11) and a voice VLAN under Cisco IOS or auxiliary VLAN under CatOS for voice traffic (represented by VVIDs 110, 111, 310, 311, and 312 in Figure 6-11).

Separate voice and data VLANs are recommended for the following reasons:

- Address space conservation and voice device protection from external networks
- Private addressing of phones on the voice or auxiliary VLAN ensures address conservation and ensures that phones are not accessible directly via public networks. PCs and servers are typically addressed with publicly-routed subnet addresses; however voice endpoints should be addressed using RFC 1918 private subnet addresses.
- QoS trust boundary extension to voice devices
- QoS trust boundaries can be extended to voice devices without extending these trust boundaries and, in turn, QoS features to PCs and other data devices
- Protection from malicious network attacks
- VLAN access control, 802.1Q, and 802.1p tagging can provide protection for voice devices from malicious internal and external network attacks such as worms, DoS attacks, and attempts by data devices to gain access to priority queues via packet tagging.
- Ease of management and configuration

  Separate VLANs for voice and data devices at the access layer provide ease of management and simplified QoS configuration.

To provide high-quality voice and to take advantage of the full voice feature set, access layer switches should provide support for:

- 802.1Q trunking and 802.1p for proper treatment of Layer 2 CoS packet marking on ports with phones connected
- Multiple egress queues to provide priority queuing of RTP voice packet streams
- The ability to classify or reclassify traffic and establish a network trust boundary
- Inline power capability (although inline power capability is not mandatory, it is highly recommended for the access layer switches)
- Layer 3 awareness and the ability to implement QoS access control lists (these features are required if you are using certain IP telephony endpoints, such as a PC running a softphone application that cannot benefit from an extended trust boundary)

Spanning Tree Protocol (STP)

To minimize convergence times and maximize fault tolerance at Layer 2, enable the following STP features:

- PortFast

  Enable PortFast on all access ports. The phones, PCs, or servers connected to these ports do not forward bridge protocol data units (BPDUs) that could affect STP operation. PortFast ensures that the phone or PC when connected to the port is able to begin receiving and transmitting traffic immediately without having to wait for STP to converge.
- Root Guard or BPDU Guard

  Enable root guard or BPDU guard on all access ports to prevent the introduction of a rogue switch that might attempt to become the Spanning Tree root, thereby causing STP re-convergence events and potentially interrupting network traffic flows. Ports that are set to errdisable state by BPDU guard must either be re-enabled manually or the switch must be configured to re-enable ports automatically from the errdisable state after a configured period of time.
- UplinkFast and BackboneFast

Enable these features where appropriate to ensure that when changes occur on the Layer 2 network, STP converges as rapidly as possible to provide high availability. When using stackable switches such as the Catalyst 2950, 3550, or 3750, enable Cross-Stack UplinkFast (CSUF) to provide fast failover and convergence if a switch in the stack fails.

- UniDirectional Link Detection (UDLD)

  Enable this feature to reduce convergence and downtime on the network when link failures or misbehaviors occur, thus ensuring minimal interruption of network service. UDLD detects and takes out of service links where traffic is flowing in only one direction. This feature prevents defective links from being mistakenly considered as part of the network topology by the Spanning Tree and routing protocols.

Note With the introduction of RSTP 802.1w, features such as PortFast and UplinkFast are not required because these mechanisms are built into this standard. If RSTP has been enabled on the Catalyst switch, these commands are not necessary.

CallManager Server Farm

Cisco CallManager cluster servers, including media resource servers, typically reside in a data center or server farm environment. In addition, centralized gateways and centralized hardware media resources such as conference bridges, DSP or transcoder farms, and media termination points are located in the data center or server farm.

The server farm is typically implemented at the access layer, which must currently be designed using traditional design methodology to the distribution layer switches because many of the features required for voice applications, such as SRST, gateways, DSP resources, and so on are not currently VRF-aware.

Because these servers and resources are critical to voice networks, Cisco recommends distributing all Cisco CallManager cluster servers, centralized voice gateways, and centralized hardware resources between multiple physical switches and, if possible, multiple physical locations within the campus.

This distribution of resources ensures that, given a hardware failure (such as a switch or switch line card failure), at least some servers in the cluster are still available to provide telephony services. In addition, some gateways and hardware resources are still available to provide access to the PSTN and to provide auxiliary services.

Besides being physically distributed, these servers, gateways, and hardware resources should be distributed among separate VLANs or subnets so that if a broadcast storm or DoS attack occurs on a particular VLAN not all voice connectivity and services are disrupted.

For more detailed information about network infrastructure requirements for a highly available, fault-tolerant campus network, see Chapter 3, "MPLS-Based VPN MAN Reference Topology" and the section QoS for Critical Applications in Chapter 4, "Implementing Advanced Features on MPLS-Based VPNs." Also see the appropriate Enterprise IP Telephony SRND at the following URL: http://www.cisco.com/en/US/products/sw/voicesw/ps556/products_implementation_design_guide_book09186a008044714e.html.

Network Services

After a highly available, fault-tolerant, multi-layer campus network has been built, network services required for IP Communications such as DNS, DHCP, TFTP, and NTP can be deployed.

It is beyond the scope of this document to address these services. For detailed design and deployment guidance for these services, see the Enterprise IP Telephony SRND at the following URL: http://www.cisco.com/en/US/products/sw/voicesw/ps556/products_implementation_design_guide_book09186a008044714e.html.

Media Resources

A media resource is a software- or hardware-based entity that performs media processing functions on the data streams to which it is connected. Media processing functions include:

- Mixing multiple streams to create one output stream (conferencing)
- Passing the stream from one connection to another (media termination point)
- Converting the data stream from one compression type to another (transcoding)
- Echo cancellation
- Signaling
- Termination of a voice stream from a TDM circuit (coding/decoding)
- Packetization of a stream
- Streaming audio (annunciation)

Note Music on hold is discussed in the following section.

There are basically no design differences with deploying media resources in the self-managed MPLS network. Media resources are normally implemented on the server farm access layer as described in Chapter 2, "Technology Overview."

For more details on media resource design, see the Enterprise IP Telephony SRND at the following URL: http://www.cisco.com/en/US/products/sw/voicesw/ps556/products_implementation_design_guide_book09186a008044714e.html.

Music on Hold

Music on hold (MoH) is an integral feature of the Cisco IP Telephony system that provides music to callers when their call is placed on hold, transferred, parked, or added to an ad-hoc conference.

Implementing MoH is relatively simple, but requires a basic understanding of unicast and multicast traffic, MoH call flows, configuration options, server behavior and requirements.

This section describes MoH at a high level. For additional configuration and design details for implementing MoH, see the Enterprise IP Telephony SRND at the following URL: http://www.cisco.com/en/US/products/sw/voicesw/ps556/products_implementation_design_guide_book09186a008044714e.html.

Deployment Basics of MoH

For callers to hear music while on hold, Cisco CallManager must be configured to support the MoH feature, which requires an MoH server to provide the MoH audio stream sources as well as the Cisco CallManager configured to use the MoH streams provided by the MoH server when a call is placed on hold.

The integrated MoH feature allows users to place on-net and off-net users on hold with music streamed from a streaming source. This source makes music available to any on-net or off-net device placed on hold. On-net devices include station devices and applications placed on hold, consult hold, or park hold by an IVR or call distributor. Off-net users include those connected through Media Gateway Control Protocol (MGCP) and H.323 gateways. The MoH feature is also available for plain old telephone service (POTS) phones connected to the Cisco IP network through Foreign Exchange Station (FXS) ports.

The integrated MoH feature includes media server, database administration, call control, media resource manager, and media control functional areas. The MoH server provides the music resources and streams.

You can configure the MoH feature via the Cisco CallManager Administration interface. When an end device or feature places a call on hold, Cisco CallManager connects the held device to an MoH media resource. Essentially, Cisco CallManager instructs the end device to establish a connection to the MoH server. When the held device is retrieved, it disconnects from the MoH resource and resumes normal activity.

Unicast and Multicast MoH

Cisco CallManager supports unicast and multicast MoH transport mechanisms.

Unicast MoH consists of streams sent directly from the MoH server to the endpoint requesting an MoH audio stream. A unicast MoH stream is a point-to-point, one-way audio Real-Time Transport Protocol (RTP) stream between the server and the endpoint device.

Unicast MoH uses a separate source stream for each user or connection. As more endpoint devices go on hold via a user or network event, the number of MoH streams increases. Hence if twenty devices are on hold, then twenty streams of RTP traffic are generated over the network between the server and these endpoint devices. These additional MoH streams can potentially have a negative effect on server CPU resources, network throughput, and bandwidth. However unicast MoH can be extremely useful in those networks where multicast is not enabled or where devices are not capable of multicast, thereby still allowing an administrator to take advantage of the MoH feature.

Multicast MoH consists of streams sent from the MoH server to a multicast group IP address that endpoints requesting an MoH audio stream can join as needed. The self-managed MPLS MAN architecture supports multicast as described in Multicast, page 6-31 and is the preferred design method in deploying MoH in this architecture.

A multicast MoH stream is a point-to-multipoint, one-way audio RTP stream between the MoH server and the multicast group IP address. Multicast MoH conserves system resources and bandwidth because it enables multiple users to use the same audio source stream to provide MoH. Hence if twenty devices are on hold, then potentially only a single stream of RTP traffic is generated over the network.

For this reason, multicast is an extremely attractive technology for the deployment of a service such as MoH because it greatly reduces the CPU impact on the source device and also greatly reduces the bandwidth consumption for delivery over common paths. However multicast MoH can be problematic in situations where a network is not enabled for multicast or where the endpoint devices are not capable of handling multicast.

For information about IP multicast network design, see Multicast, page 6-31.

Recommended Unicast/Multicast Gateways

The following recommended gateways support both unicast and multicast MoH:

- Cisco 6624 and 6608 gateway modules with MGCP and Cisco CallManager Release 3.3(3) or later
- Cisco Communication Media Module (CMM) with MGCP or H.323 and Cisco CallManager Release 4.0, Cisco IOS Release 12.2(13)ZP3 or later, and Catalyst OS Release 8.1(1) or later

- Cisco 2600, 2800, 3600, 3700, and 3800 Series Routers with MGCP or H.323 and Cisco IOS Release 12.2(8)T or later

MoH and QoS

Convergence of data and voice on a single network requires adequate QoS to ensure that time-sensitive and critical real-time applications such as voice are not delayed or dropped. To ensure proper QoS for voice traffic, the streams must be marked, classified, and queued as they enter and traverse the network to give the voice streams preferential treatment over less critical traffic. MoH servers automatically mark audio stream traffic the same as voice bearer traffic with a Differentiated Services Code Point (DSCP) of EF (ToS of 0xB8). Therefore as long as QoS is properly configured on the network, MoH streams receive the same classification and priority queueing treatment as voice RTP media traffic.

Call Processing

Call processing is a critical component of IP Communications design. There are no changes in call processing design when deploying CallManager in a self-managed MPLS environment versus traditional enterprise design other than the deployment models that are described in IP Telephony Deployment Models over the Self-Managed MPLS MAN, page 6-8.

For detailed design guidance for IP Communications call processing, see the Enterprise IP Telephony SRND at the following URL: http://www.cisco.com/en/US/products/sw/voicesw/ps556/products_implementation_design_guide_book09186a008044714e.html.

Cisco Unity Messaging Design

This section focuses at a high level on the design aspects of integrating Cisco Unity with Cisco CallManager in the various deployment models over the self-managed MPLS MAN. The design topics covered in this section apply to both voicemail and unified messaging deployments.

This section does not discuss the full design details of integrating Unity with CallManager because this is fully documented in the IP Communications call processing section of the Enterprise IP Telephony SRND available at the following URL: http://www.cisco.com/en/US/products/sw/voicesw/ps556/products_implementation_design_guide_book09186a008044714e.html.

For additional design information about Cisco Unity, including integrations with other non-Cisco messaging systems, see the Cisco Unity Design Guide at the following URL: http://www.cisco.com/en/US/partner/products/sw/voicesw/ps2237/products_implementation_design_guide_book09186a008022f63b.html.

Messaging Deployment Models

Cisco Unity supports three primary messaging deployment models in the self-managed MPLS MAN:

- Single-site messaging
- Multi-site deployment with centralized messaging
- Multi-site deployment with distributed messaging

Deployments involving both Cisco CallManager and Cisco Unity use one call processing model for Cisco CallManager and one messaging model for Cisco Unity. The messaging deployment model is independent of the type of call processing model deployed.

In addition to the three messaging deployment models, Cisco Unity also supports messaging failover.

All messaging deployment models support both voicemail and unified messaging installations.

Single-Site Messaging

In this model, the messaging systems and messaging infrastructure components are all located at the same site, on the same highly available LAN. The site can be either a single site or a campus site interconnected via high-speed metropolitan area networks (self-managed MPLS MAN). All clients of the messaging system are also located at the single (or campus) site. The key distinguishing feature of this model is that there are no remote clients across the WAN.

Centralized Messaging

In this model, similar to the single-site model, all the messaging system and messaging infrastructure components are located at the same site. The site can be one physical site or a campus site interconnected via high-speed MANs. However unlike the single-site model, centralized messaging clients can be located both locally and remotely across the WAN.

Distributed Messaging

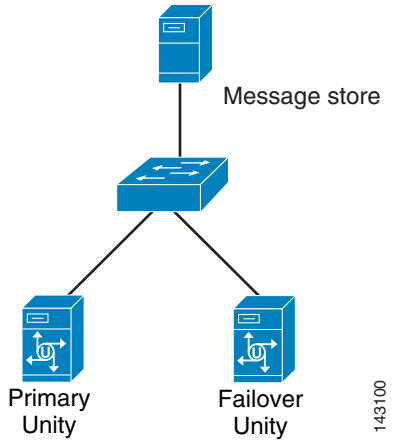
In distributed messaging, the messaging systems and messaging infrastructure components are co-located in a distributed fashion. There can be multiple locations, each with its own messaging system and messaging infrastructure components. All client access is local to each messaging system and the messaging systems share a messaging backbone that spans all locations. Message delivery from the distributed messaging systems occurs via the messaging backbone through a hub-and-spoke type of message routing infrastructure.

No messaging infrastructure components should be separated by a WAN from the messaging system they service. Distributed messaging is essentially a multiple, single-site messaging model with a common messaging backbone.

Messaging Failover

All three messaging deployment models support messaging failover. You can implement local messaging failover as illustrated in Figure 6-12. With local failover, both the primary and secondary Cisco Unity servers are located at the same site on the same highly available LAN.

Figure 6-12 Local Failover of Cisco Unity Messaging



Failover of Cisco Unity Messaging Across the MAN

At present, any configuration that requires failover of Unity across the MAN, such as the Primary Unity server at Data Center 1 and the Failover Unity Server at Data Center 2, requires review by the UBCU TME team.

The UCBU TME team is currently in the process of testing and incorporating Unity Failover across the MAN into the new Unity design guide which will be available shortly.

Cisco Unity and Cisco CallManager support the following combinations of messaging and call processing deployment models:

- Single-site messaging and single-site call processing
- Centralized messaging and centralized call processing
- Distributed messaging and centralized call processing
- Centralized messaging and distributed call processing
- Distributed messaging and distributed call processing

For further details on site classification and a detailed analysis of supported combinations of messaging and call processing deployment models, see the Cisco Unity Design Guide at the following URL: http://www.cisco.com/en/US/partner/products/sw/voicesw/ps2237/products_implementation_design_guide_book09186a008022f63b.html.

Multicast

This section describes how to add multicast VPN as an overlay service on top of an MPLS Layer 3 VPN service. The following major sections are discussed:

- Multicast VPN Service Overview
- Multicast VPN Service Architecture
- Multicast VPN Service Design and Deployment Guidelines
- QoS for mVPN Service
- Multicast VPN Security

- Design Choices for Implementing mVPN
- Multicast VPN Service Management
- Implementing and Configuring the mVPN Service

Multicast VPN Service Overview

The multicast VPN feature in the Cisco IOS software provides the ability to support the multicast feature over an MP-BGP Layer 3 VPN. This service allows users to leverage their infrastructure to deliver multicast with minimal investment.

The Cisco implementation of mVPN offers these benefits:

- The implementation of a data Multicast Distribution Tree (MDT) that allows for a scalable delivery of traffic
- Several PIM options in the core for both data and default MDTs. SSM is offered as an alternative to Anycast RP or Auto-RP
- Support for a broad range of multicast options within the VPN including Anycast RP, Auto-RP, Static RP, Bi-Dir, and accept-register filters in the VPN
- Support for Rendezvous Point (RP) on a PE on a per-VRF basis, VRF awareness of multicast related MIBs, the Cisco mVPN MIB, and MSDP in the VPN (for RP redundancy and RP management service)

Multicast VPN Service Architecture

Service Components

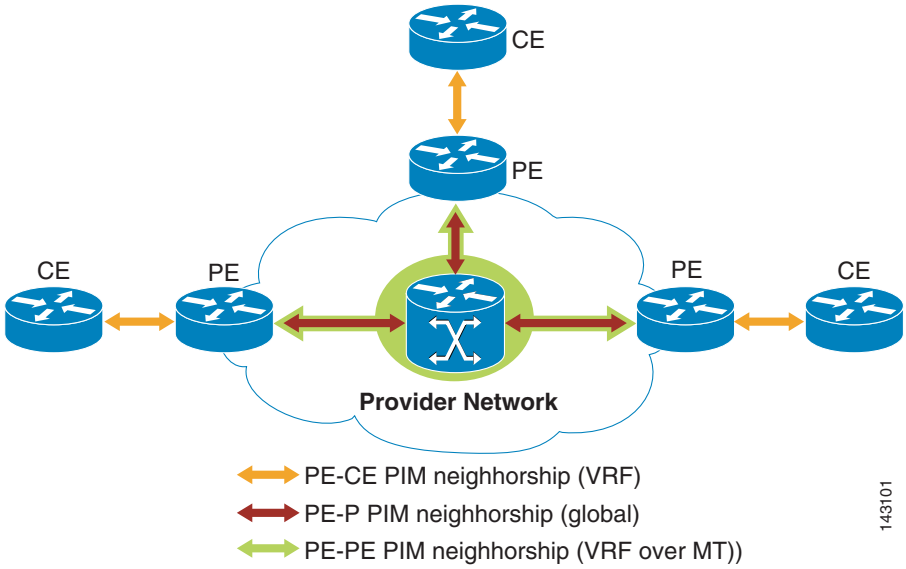
Multicast VPN (mVPN) is an overlay service to the MPLS Layer 3 VPN service that provides multicast forwarding support to VPN users. However mVPN does *not* depend on MPLS. An IP multicast-enabled core is the only requirement. The mVPN feature needs multiprotocol BGP (MP-BGP) VPN support. Therefore mVPN can be implemented as an overlay service on any VPN built with MP-BGP or even without any unicast VPN service.

The following are the required functionalities for the mVPN service:

- Underlying IP routing in the core and in the VPN
- Multicast support in the core
- Underlying MP-BGP-based VPN service
- Multicast VRF support on the PEs and MTI
- Multicast support in the VPN (PE interfaces facing the CE)

Figure 6-13 shows the mVPN service architecture.

Figure 6-13 mVPN Service Architecture



The mVPN service allows for the support of the user multicast traffic in an MP-BGP VPN environment; therefore allowing for support of multicast video, voice, and data within the VPN. The current Cisco implementation of multicast VPN supports:

- Multiple user-facing multicast protocol options (PIM-SM, PIM-DM, SSM, PIM-Bidir, BSR, static or auto RP, and SPT threshold values)
- Multiple multicast core options for the MDT support (PIM-SM, SSM, PIM-Bi-Dir)
- Optimization of bandwidth in the core using configurable data MDTs

The following subsections describe the different service components that should be implemented on the network architecture to provide the mVPN service.

Multiprotocol BGP

Multiprotocol Border Gateway Protocol (MP-BGP) offers a method for network administrators to distinguish which prefixes to use for performing multicast reverse path forwarding (RPF) checks. The RPF check is fundamental in establishing multicast forwarding trees and moving multicast content successfully from source to receiver(s). MP-BGP is based on RFC 2283 multiprotocol extensions for BGP-4.

MP-BGP is also used to propagate the MDT information that allows the PIM neighborships between PE routers to be created and to set up the multicast tunnels for the VPN. The MDT group address is carried as a VPNv4 address along with MDT source information (BGP peering source address).

New Extended Community Attribute

A new extended community attribute has been created to carry MDT source information along with the MDT group address in the BGP updates.

MVRF

Multicast VPN routing and forwarding instance (MVRF) is a virtual forwarding table that supports multicast entries. This feature needs to be enabled on the router for the VRF to support the mVPN service.

Multicast Tunnel Interface (MTI)

Each MDT is associated with an MVRF and the interface that points to this MDT (from the MVRF) is an MTI. The MTI is listed in the outgoing interface list (OIL) for the VRF multicast-group entries. MTI is a virtual interface that gets automatically created when the default MDT is configured in the VRF context.

Multicast Domain (MD)

An MD is a collection of MVRFs that can exchange multicast traffic. Effectively, every PE router in a multicast domain becomes both a sender and a member of the corresponding MDT group.

Multicast Distribution Tree (MDT)

MDT groups are used to encapsulate and transport VPN traffic within the corresponding multicast domain. Multicast routing states for MDT groups are created and maintained in the global routing table only. To a P router, an MDT group appears the same as any other multicast address. It does not require any special handling for routing and forwarding. However because VPN packets need to be encapsulated and decapsulated as they enter and exit (respectively) the multicast domains, a PE router handles MDT groups differently.

A mesh of point-to-multipoint multicast tunnels are established for each multicast domain. At the root of the tunnel is a PE router. The leaves are also PE routers that have MVRFs in the same multicast domain. Typically, there is a single MVRF in a MDT.

There are two types of MDTs:

- Default MDT—Tree created by the mVPN configuration. The default MDT is used for user control plane and low rate data plane traffic, as well as any dense-mode traffic. It connects all of the PE routers with the MVRF in a particular MD. A default MDT exists in every mVPN whether or not there is an active source in the user network.
- Data MDT—Tree created dynamically by the existence of active sources in the user network sending to active receivers located behind separate PE routers. The tree is triggered by a traffic-rate threshold. Data MDT connects only PE routers that have active sources or receivers of traffic. The data MDT is created only by (S,G) Sparse or SSM state in the user mVPN network and is never created for dense-mode (S,G) state.

Multicast VPN Service Design and Deployment Guidelines

The following features are required to support the mVPN service:

- Core and PEs (on the core-facing and loopback interfaces) need to be multicast-enabled (PIM-SM/SSM/BIDIR) and support the different RP mapping options when applicable
- PEs need MP-BGP support for multicast
- PE needs to support GRE and multicast tunnel interface (MTI) for MVRF binding

- PE edge-facing interfaces need to support the user multicast functionalities (PIM type and RP mapping options or the SSM or BiDir range defined)

The following additional features are highly recommended for performance, scalability, and user-specific requirements:

- Support for RR functionality to distribute MP-BGP multicast-related information between PEs
- Support for core optimization using data MDTs
- Hardware-accelerated forwarding
- Management support—mVPN MIB and VRF-awareness of relevant multicast MIBs

Note SSM support in the core is recommended. The following address range is the Internet SSM space: 232.0.0.0 through 232.255.255.255. If SSM is deployed in the core, it is recommended to extend this space into some private SSM space (part of administratively scoped addresses: 239.0.0.0-239.255.255.255); thus the Cisco recommendation is 239.232.X.X. This requires that the **ip pim ssm range** command *not* use the default keyword, because this uses the 232.0.0.0 address space for the data MDTs.

Service Deployment

Enabling mVPN on PE Devices that Support VPN Services

To overlay a multicast VPN service on top of an existing VPN service, the following steps are required:

Step 1 Choosing the PIM mode for the core network.

Cisco recommends PIM-SSM as the protocol in the core. For source discovery, a new attribute is added to BGP , so no additional configuration is required.

A new RD type is used to advertise the source of the MDT together with the MDT group address. PIM SM has been the most widely deployed multicast protocol and has been used for both sparsely and densely populated application requirements. PIM SSM, although newer, is based on PIM SM without an RP. In SSM receivers immediately join the Shortest Path Tree back to the source without joining the RP shared tree. Either PIM SSM or PIM SM are suitable for the default MDT or the data MDT.

PIM SSM is simpler to deploy than PIM SM. It does not require a Rendezvous Point and the core network is a known and stable group of PE multicast devices. Cisco recommends the use of PIM SSM for mVPN core deployments.

Step 2 Choosing the VPN group addresses used inside the core network.

In Step 1, PIM-SSM was selected. The default PIM-SSM range is 232/8; however this address range is designed for global use in the Internet. For use within a private domain, the use of an address out of the administratively scoped multicast range, 239/8, is recommended (RFC2365). Using this private address range makes it simpler to filter on boundary routers.

Cisco recommends using 239.232/16, as this address range is easily recognizable as both a private address and a SSM address by using 232 in the second octet.

In the design discussed in this document, the range is divided for default-MDT and data-MDT. Data-MDT is discussed elsewhere in this document.

Default-MDTs uses 239.232.0.0-239.232.0.256 and Data-MDTs use 239.232.1.0-239.232.255.255. This address range provide support for up to 255 MVRFs per PE router.

Step 3 Configuring the core network for PIM-SSM.

The following commands need to be configured to enable a basic PIM-SSM service.

a. On all P and PE routers configure globally:

```
ip multicast-routing
access-list 1 permit 239.232.0.0 0.0.255.255
ip pim ssm range 1
```

b. On all P interfaces and PE interfaces facing the core configure:

```
ip pim sparse-mode
```

c. On the PE routers configure on the loopback interface used to source the BGP session

```
ip pim sparse-mode
```

Step 4 Configuring the MDT on the VRF.

a. To configure multicast routing on the VRF, configure on all PE routers for the VRF:

```
ip vrf <vrf-name>
mdt default 239.232.0.1 (I just don't like 0.0)
mdt data 239.232.X.0 0.0.0.15 threshold 1
```

b. Choose a unique data MDT address (X) for each VRF that supports IP multicast.

To enable multicast routing for this VRF, configure the following:

```
ip multicast-routing vrf <vrf-name>
```

c. Choosing the PIM Mode for the VPN:

The PIM mode inside the VPN depends on what the VPN edge network is using. Cisco provides automatic discovery of Sparse Mode RPs inside a VPN via auto-rp or bsr, which requires no additional configuration. Static RPs, Anycast RPs, and Dense Mode PIM are also supported on a per MVRF basis.

Optionally, a PE router can be configured as the RP on a per VRF basis.

On the PE, define the RP used within the VPN:

```
ip pim vrf <vrf-name> rp-address <ip address> override
```

Step 5 Configuring the PIM mode inside the VPN.

Configure all PE-CE interfaces for sparse-dense-mode that ensures that either auto-rp or bsr messages are received and forwarded allowing the PE to learn the group to RP mapping inside the VPN. Use sparse-mode for static RP or Anycast RP deployments.

a. Configure on all edge facing interfaces:

```
ip pim sparse-dense-mode
```

Or

```
ip pim sparse-mode
```

The following is a sample of an edge-facing network configuration:

```
interface Ethernet0/0
 ip vrf forwarding <vrf-name>
 ip address 20.0.1.1 255.255.255.0
 ip pim sparse-mode
!
```

The following is a configuration example:

```
ip vrf blue-data
 rd 10:105
 route-target export 10:105
 route-target import 10:105
 mdt default 239.232. 0.10
 mdt data 239.232.10.0 0.0.0.15 threshold 1
ip multicast-routing distributed
ip multicast-routing vrf blue-data distributed

ip pim vrf blue-data rp-address 1.1.1.11 override
ip pim vrf blue-data ssm range 1
access-list 1 permit 239.232.0.0 0.0.255.255
```

The format of the default MDT utilizes 239 as the first octet as a private address, 232 as the second octet to identify it as using PIM SSM, while the third octet can be used to relate the MDT to the VRF that it supports to provide for ease of troubleshooting.

Multicast Core Configuration—Default and Data MDT Options

Selecting Protocol for the MDTs

The selection of the multicast protocol (PIM mode) for the MDTs can be done using access lists that can be configured when defining default and data MDTs in the MVRFs. ACLs define the address range to which a given multicast protocol is assigned. Therefore, depending on the VPN user requirements, different protocol options can be used for each MDT. To minimize complexity, it is recommended that the same PIM deployment options be used for all mVPN MDTs.

Protocol Design Options for Default MDT

Generally, Cisco recommends that SSM be implemented in the MPLS core for all MDTs. SSM allows for a simpler deployment, avoids any RP administration or management, eliminates the RP as potential point of failure, and offers efficiency (does not depend on the RP or shared trees).

In the cases where the users might want to keep their existing multicast configurations such as PIM-SM, using Anycast RP, Auto-Rp, or static RP, they can implement any PIM mode for the default and data MDTs. The support for Bidir in the core also allows for simplicity of the control plane (no SPTs) and scalability (eliminates all [S,G] mroute states in the core because it only uses [*,G]), which makes PIM-BiDir an ideal choice for the default MDT as the default MDT carries all control traffic between PEs. BiDir makes sense for very large scale mVPN deployments to limit the multicast route state maintained in the MPLS core of the network.

Protocol Design Options for RP Positioning

In the case of PIM-SM and BiDir, at least one RP is required in the core multicast network for the MDT. Because traffic traverses the RP shared tree (*,G), the RP should be placed in a centralized location and should not be placed on a PE to avoid additional load on the edge routers.

Addressing Ranges and Considerations

Multicast Core—MDT Addressing

Each multicast domain is assigned a distinct group address from a pool of multicast addresses. The group ranges reserved for multicast domains are called MDT groups in this document. The recommended group range for multicast domains is a subset of the administratively scoped addresses: 239/8. The administratively scoped range of addresses is a private address space that is not meant to be exchanged on the Internet.

Note that MDT group addresses are global in the core; each VPN has a unique set of MDT addresses (default and data). Therefore, the addresses cannot overlap if PIM-SM is used.

There is a specific range for Internet SSM (232.0.0.0 to 232.255.255.255). It is not recommended to select from these addresses for SSM default and data MDTs because they are public Internet SSM group addresses. Cisco suggests using 239.232/16 for private SSM addresses. Note that the choice of SSM for MDTs in the core allows optimization of the address space as a given group address may be used by several sources. With SSM each PE MDT has a unique (S,G) even if the same group address is used by different PEs.

When configuring the mVPN, ensure that all PE routers of a given VPN have the same default MDT group.

The setup of MDT and creation of the MTI are conditioned by certain addressing parameters for the MP-BGP update source and PIM neighbor addresses. It is important to set the proper addressing for the mVPN to be up:

- The MTI takes its properties from the interface used for the BGP peering (MP-BGP), usually defined as a loopback address using the *update-source* parameter of BGP.
- The RPF check done for the PIM neighborship also uses the BGP information (next hop) as no unicast routing protocol runs on the MTI. Therefore the PIM peer ID and BPG next-hop addresses of a PE must match. Note that this next hop must be reachable by the core P routers for RPF check.
- The BGP peering address is also used as the source address of the local PE for the multicast tunnels (root of the default and data MDTs).

VPN Network Multicast Addressing

There are no restrictions on the addressing within a VPN because a private address space is provided per definition of the service. VPN address spaces may overlap unless extranets are configured or Internet access is provided.

Caveats

Some other restrictions in addition to the platform-specific issues mentioned in the sections above include:

- If the core multicast routing is using source-specific multicast (SSM), then the data and the default MDT groups must be configured within the SSM range of addresses. Cisco recommends using a range of 239.232/16.
- Data MDTs are not created for VRF PIM dense-mode multicast streams. PIM DM relies on a tree that reaches all CE routers; therefore, the default MDT is used.

- Multiple BGP update sources are not supported and configuring them can break the mVPN RPF check. The source IP address of the mVPN tunnels is determined by the highest IP address used for the BGP peering update source. If this IP address is not the IP address used as the BGP peering address with the remote PE router, mVPN does not function correctly.

QoS for mVPN Service

Because the mVPN feature is generally an additional service provided along with Layer 3 VPN unicast, users expect the same QoS and flexibility. This specifically applies to the transport of their time- or business-sensitive multicast traffic that requires prioritized transport and low drop rate. MoH and video streaming are examples of multicast applications that benefit from QoS.

The QoS requirements include:

- The existing QoS transparency models (pipe, short pipe, or uniform) for unicast MPLS VPN traffic should be enhanced to support mVPN traffic.
- All the ingress and egress QoS policies on all the CE, PE, and P routers should be modified to support multicast or mVPN traffic.
- Statistical reporting and measurements need to provide accurate information for mVPN traffic.

Implementing QoS for the mVPN

Baseline IP performance degradation or failure naturally impact the quality of service offered for mVPN. Note that the mVPN traffic is GRE or IP in IP traffic sent over the multicast core. Therefore it is a good practice to ensure initial good performance and reliability of the IP core and the mVPN service itself. Other factors that influence QoS include convergence, scalability, and performance.

Implementing QoS for mVPN on the PE Routers

PE routers are the devices that are the most critical points in terms of performance and are also the entry point for the multicast VPN user traffic. Therefore most aspects of QoS implementation are recommended on the PEs, depending on platform support.

Recall that the PEs perform most of the service features (mVPN encapsulation, decapsulation, and lookups), keep the VPN multicast information (MVRF, mstates in the global routing table for data and default MDTs), and cumulate this information with regular unicast route and control plane information. The PE routers are also the point where other services are implemented (for example, security and 6PE functionalities).

Implementing QoS for mVPN on the P Routers

QoS support in the core implies enabling QoS in the core for IPMC traffic, as the multicast VPN packets are sent as IP-in-IP or GRE-encapsulated multicast-traffic over the MPLS core.

Implementing QoS for mVPN on the CE Routers

The CE could be the place in the network where classification and potential marking or remarking of packets would be done. This offloads the PE. If this is not done, then the multicast packets arrive at the ingress PE interfaces marked according to the user classes of services and might need policing, class mapping, and remarking.

Multicast VPN Security

The VPN multicast feature in itself is a secure service ensuring the transport of multicast information. Per definition, a VPN service must guarantee privacy of the user traffic. Security mechanisms should be implemented from the very beginning of a service offering because they are critical to service definition, assurance, and SLAs fulfillment.

Security of the mVPN service can be considered at two levels:

- Protection of resources—This ensures that a mVPN service is up and functioning; i.e., transporting the user information. The resources to be protected are the CPU load and memory usage of routers (Ps, PEs, CEs, RRs), link bandwidth protection and guarantee, and resource access protection (physical and administrative-login). The resources of the core and the edge are shared by several mVPN users; therefore a node or link failure would have an impact on the service of several users.
- Protection of the user information (privacy, integrity)—This means protecting the access to the control and information flows (permission to be a receiver for a group or an MDT, permission to connect to a VPN and receive VPN multicast data,and isolation of flows between different user groups),and control of information sources and access to the VPN (on the PE-CE and access to the MDT resource).

The different security design recommendations for the multicast VPN service are further divided according to the following four types of features, each of which might present resource protection and user information security levels:

- Access control—Which sources may send multicast data (in the VPN or in the core)
- Admission control—Who receives the multicast data (receivers in the VPN or MDT members)
- Data plane control—Protecting the data exchanged over the mVPN service
- Control plane control—Protecting the control information

Implementing Security for mVPN Service

Several important levels of security should be implemented on the PE, which is an important element of mVPN service security because it is located between VPNs and the MPLS core network and performs the most mVPN-relevant operations. Therefore it is exposed to a high control plane and data traffic load. Regarding resource protection, you want to limit the control plane memory and CPU usage (number of VPN multicast routes, cycles to generate trees, cycles to maintain BGP information, and so on) as well as the data plane load (unicast and multicast traffic forwarding, double lookup, and encapsulation or decapsulation of the mVPN traffic). It is important to control access to the multicast information flows and the control plane itself to avoid intrusions, protect integrity, and to respect the separation of the traffic between mVPN service users.

In the case of resource protection, the mVPN service requires a careful design of the core in terms of the number of MDTs and multicast VPN routes. These factors impact the load on the P routers (amount of mstates and traffic load). When using PIM-SM or BiDir in the core, the RP functionality and placement should be carefully considered for redundancy, access control (that is, which PE accesses an MDT as a source/receiver), and control plane load.

Levels of security that can be implemented on the CE router include:

- Router access
- Control and limitation of the multicast data sent to the PE, which reduces load and risks of DoS attacks and ensures core and PE scalability
- Limitation of the resources used by the control plane on the CE

Care should be taken when configuring the RP parameters to ensure controlled source access and receiver access to the VPN multicast groups, to restrict control plane load because of the user-facing RP functionality, and to implement redundancy for the RP service (RP service feature assurance) because the PE is the most exposed component of the mVPN service.

Note that all unicast Layer 3 VPN security recommendations are also relevant in the case of an mVPN service; the PE and core are shared by unicast and multicast VPN traffic because both types of traffic are transported on the same infrastructure.

Access Control

Source access is an important security aspect of multicast in general and applies to an mVPN service as well for two reasons: it affects the privacy guarantee of a VPN and it is a resource protection measure, protecting the user resources as well as the core and edge multicast resources. A rogue source can generate a lot of traffic and overwhelm receiver and network resources along the trees, which can be a result of DoS attacks.

PIM-SM and BiDir

PIM-SM and BiDir use RPs for source registration. The following features allow access control of sources registering either at the multicast core RP or a VPN user network RP level:

- Source registration

  To define access lists and route maps to filter incoming register messages based on the source address or (S,G) pair, use the source registration feature configured on the RP themselves. This feature defines which sources or (source, group) parameters are accepted while the rest are rejected. It is a global or VRF aware command (**ip pim accept-register [list <acl>][route-map <>map]**). Use this feature when the sources and multicast groups are known and when the RP is close to the sources.

  Certain considerations should be taken into account. The first hop routers located between the source and the RP still receive the traffic from any source. As a result, this traffic creates control and data workload on these routers as they process packets and generate corresponding mstates. The source-register filter within a VPN can be used to limit which traffic is accepted as multicast VPN traffic; therefore limit the traffic across the multicast core (on the MDT). In the core, because the network knows the MDT group addresses to expect, this feature allows unexpected multicast group traffic (MDT or none) over the multicast core to b e denied. For example, imagine that a rogue device sending traffic from a source to an MDT group, thereby becoming an illegitimate source on the VPN. This is possible with multicast protocols such as PIM-SM and BiDir where an RP is being used. A source sending to the RP is requested to only have a PIM neighborship with the RP and to register with the RP. This is possible because PIM does not currently have any authentication mechanisms. One way to prevent this from happening is to set appropriate source filters on the RP as described above.

- Local loopback RP

  An alternative to source access control when using RPs is to implement the local loopback RP method that is used to blackhole locally the traffic sent by rogue sources towards a multicast group (**ip pim rp-address <local_loopback> <acl>**). This command requires the user to explicitly define which group addresses are not authorized. This is less restrictive than the previous option in the sense that an undefined source is able to send if not listed in the access list. The feature has the advantage to avoid rogue traffic even to the first hop routers located on the path between local router and the RP. This avoids control workload on these routers but requires local configuration on each router and cannot prevent load on the very first router and multicast to local sources. This is more efficient in terms of first hop routers protection; however the above limitations should be kept in mind.

- **ip pim accept-rp**

  The **ip pim accept-rp** is used by routers to select which RPs are accepted for a given set of multicast groups. This feature can be applied within the VPN and in the core. In the core it prevents a rogue RP from forwarding information on the MDTs in the cases of PIM SM and BiDir in the core. Note that the MDT group addresses are well known because they are predefined (default MDT address plus a range of addresses for data MDTs). Therefore it is simple to define the groups for which a given RP set is accepted or rejected.

The previous features emphasize that previous knowledge of VPN user traffic patterns and a proper addressing scheme in the core are important design recommendations when considering access control security. See Addressing Ranges and Considerations, page 6-38 for addressing recommendations.

PIM-SSM

In the case of SSM, which does not use RPs, source control is implicitly implemented because receivers must have a prior knowledge of a source to request for their local router to join the corresponding channel (use of [S,G] versus [*,G]). This presents a certain level of protection against the potential threat of DoS and VPN privacy attacks from rogue sources and can be used in the VPN user network (PE-CE and beyond) as well as in the core.

Access Lists

Any mVPN packet belonging to a user multicast flow is sent site-to-site over the default or data MDT. The VPN user network sources are thus effectively sources of multicast core traffic. In the absence of filtering everything is flooded over the default MDT or sent over the data MDT. Even if data MDT creation and access can be restricted, traffic is sent over the default MDT and affects the core. The following two practices are recommended to protect the multicast core:

- ACLs on the PE-CE should be used to limit the VPN user multicast traffic that travels site-to-site over the MDTs. This preserves the core resources, avoids some DoS attacks, and preserves PE resources by avoiding unnecessary processing of multicast traffic. Some knowledge of VPN user sources and group addresses is necessary to implement these ACLs. Using data ingress ACLs is highly recommended because it preserves network resources as well as PE resources.

Note If ACLs are configured on the PE-CE link for security in conjunction with the mVPN feature, a decrease in performance load might be noticed because of the scanning of packets.

- Implementation of QoS can provide core resources protection because traffic policing and to some extent traffic shaping and congestion avoidance can limit the traffic sent over MDTs (policing based on VPN user addressing at the ingress, WRED and shaping based on MDT group address at the egress). In these cases, the use of QoS allows one to protect the core resources.

Admission Control

Receiver access to the multicast traffic is an important VPN flow privacy concern. Within the VPN, receiver access to a multicast group or to the traffic sent by a specific source for a group can be limited using IGMP access groups. This is a security feature to be implemented within the VPN. It is configured on the PE if any host is expected to connect directly to the PE router.

Regarding resource protection on the PE, use of the IGMP access groups limits the load of the PE to a certain extent because a PE subscribes only to a data MDT if there is a receiver requesting the corresponding (S,G) traffic.

The join of a rogue P to the MDT tree as a leaf only necessitates a PIM neighborship with the RP or one of the core routers. Because PIM does not currently support peers authentication, this is a potential security issue for the mVPN service. However note that this necessitates physical connectivity access to one of the core or edge routers.

Control Plane Protection

For control plane, physical and login access to the devices (PEs, CEs, Ps, and RRs) and links should be protected. Also keep in mind that multicast uses unicast routing information; therefore security of the control plane at this level should also be considered.

BGP Neighbors

Existing BGP neighbor authentication mechanisms (MD5 authentication) for PE-PE or PE-RR information-exchange can be used.

PIM Neighbors

Overall, an intelligent design of the PIM options (see below for RP choices) for the core can potentially reduce the risks of attacks or failure, therefore providing a good resource protection design, redundancy (RRs, RPs, and sink RP in auto-RP cases), and careful selection of thresholds (SPT). The **ip pim neighbor-filter** *acl* command allows you to administratively deny a PIM neighbor from participating in PIM and is a first level of protection. To set up the PE-to-PE PIM adjacencies, the PIM router ID must be the same as the BGP peering address, which presents some level of protection against rogue PE routers.

Limiting mroutes

An important factor in terms of mVPN support is the number of mroute states in the core and on the PEs, which is a potential door for DoS attacks. Limitations of the number of routes exchanged among BGP peers and of routes installed per VRF are two ways to limit the impact of this type of attacks:

- Limitation of the number of multicast routes in an MVRF—The **ip multicast [vrf vrf-name] route-limit limit [threshold]** command limits the number of multicast routes that can be added to a multicast routing table, thus countering any attempt to fill up multicast routing states. From a service assurance and monitoring perspective, a threshold can be created and an error message is created when the limit is exceeded. The message recurs until the number of mroutes drops under the limit set by the limit argument.

Note The **max route** option of the VRF setup is applied to unicast routes only and cannot be leveraged for multicast flows.

- Limitation of the number of multicast routes accepted from a BGP neighbor—The **max route** option per-BGP neighbor can be used to limit mroutes received from a neighbor. The following example shows how to configure the maximum value for the IPV4 address family, which applies to IPV4 unicast and multicast routes:

```
pop2-7206-11-PE(config-router-af)#neighbor <IP@> maximum-prefix <MaxPrefixValue>?
<1-100> Threshold value (%) at which to generate a warning msg
restart Restart bgp connection after limit is exceeded
warning-only Only give warning message when limit is exceeded
<cr>
```

Limiting mstates

Limiting the impact of VPN user traffic on the creation of data MDTs triggered by traffic sourced in the VPN network is an important aspect of core resources protection (data MDT creation process is dynamic and requires core CPU and memory resources). The following features achieve this:

- The data MDT command (for example, mdt data 239.232.1.0 0.0.0.255 threshold 50 list 5 of the mdt data MDT_data_add wildcard threshold value_thresh list acl_# command) provides some level of protection because an access list defines the type of VPN user multicast traffic that triggers the creation of a data MDT (an ACL can define a group address or a source and group address). Cisco recommends carefully defining the VPN sources and groups that may trigger the creation of a data MDT. Note that this implies that the core must be configured for support of the multicast sources IP addresses of the user.
- For a given VPN, the amount of MDTs is limited by default to 256 (255 data MDTs and one default MDT) and can be configured to a lower value. Therefore, the impact of having potential sources creating a DoS threat because of high control plane activity and memory usage is limited by this implementation option. The 255 limitation provides some level of protection of core resources from a control plane perspective and should be used together with other resource protection features. However note that DoS attacks can take place because high rate sources send over MDTs and potentially use up multicast core resources.

The choice of data MDT threshold may also be an effective element of PE and P resource protection because its value defines which amount of traffic triggers the creation of a data MDT and therefore restricts the use of control plane resources for the dynamic setup of the MDT and the mroute state maintenance in the core.

As a general good practice with respect to data MDT monitoring, use the following command in vrf mode, which allows data MDT reuse to be logged:

```
Router(config-vrf)# mdt log-reuse
```

Note The feature enabled with the **mroute state limit** command per-interface and per-VRF is currently designed to provide resources protection and prevent potential DoS attacks (for example, the limitations of mroute states received per interface). It is used to limit the number of mroutes accepted from the PIM neighbor.

RP Protection

When RPs are configured in the core or in the VPN, Cisco recommends using redundancy mechanisms, depending on your RP mode, such as:

- Anycast RP using MSDP in the case of static RPs
- Mulitple candidates and mapping agents for auto-RP
- Multiple candidates in case of BSR, as well as redundancy of the bootstrap router

The RP functionality may be CPU-intensive on a router so care should be taken when choosing which routers support this functionality.

If auto-RP is used, use the **ip pim rp-announce-filter rp-list group-list** command on the mapping agent to restrict the routers that are accepted as RP candidates for a specific group range. This feature is recommended in the core and at the VPN network level to filter out false C-RPs. On the RPs themselves, when using MSDP, Cisco recommends using the **ip msdp SA-filter** and **ip msdp SA-limit** features to filter incoming SAs from a peer and to protect against DoS attacks by limiting the global number of SA messages on the local MSDP peer.

Overall, when designing the service and applying security-oriented features, some features might incur additional processing at the control of data plane. The impact of these features on the overall performance of the service should be kept in mind.

Design Choices for Implementing mVPN

Multicast Address Ranges for the MDTs in the Core

To provide resource protection for more scalability and to accommodate several levels of service and types of requirements, it is possible to use multicast address ranges to offer different core multicast support options to different users. For example, for a given VPN user group A that is associated with the VRF mcast1 and with the default MDT address 239.232.0.1, the data and default MDTs traffic is carried in the core using PIM SSM; for user group B that is associated with the VRF mcast2 and with the default MDT address 239.232.0.2, the default MDT uses PIM-BiDir and the data MDT uses SSM. Note that options such as RP mode (static, auto, and BSR) and the RP router choice can be also configured per user group at the MDT level using address ranges.

Data MDT Implementation Options

The following are data MDT implementation options:

- An advantage of using data MDTs is the ability to optimize flows across the P network. This offers core optimization to delivery scalability and an optimized use of core resources (memory and CPU) because it reduces the load on the PEs in an mVPN. If the user has multicast data streams with heavy throughputs or one-to-many type of streams, the use of data MDTs is highly recommended. Data MDTs are triggered at the PE closest to the source. PEs with interested receivers send dynamic “joins” to the data MDT after they receive a receiver IGMP report for the associated group. The setup of the data MDT itself creates additional states in the core and generates CPU load on the PEs. It might not be necessary to set up data MDTs for streams whose sources reach almost all VPN sites (remote PEs) or those with a very low throughput.
- Access to the data MDT—Address ranges can be used to regulate which source multicast addresses may trigger the creation of a data MDT. This is a good practice (remember the trade-off for data MDTs: additional states created in core network for a more efficient bandwidth usage) and it also allows a way to attribute certain group multicast prefixes to certain VPN multicast address prefixes for the data MDT when user stream addresses are well known.
- The value of the data MDT threshold is an important parameter for the core stability and scalability. It can be configured per data MDT group address range in a VPN and customized per-user VPN multicast stream (defined using access lists).

Convergence for the mVPN Service

Dependencies

Multicast VPN does not use MPLS switching in the core. Convergence depends only on routing protocol and multicast parameters. An MPLS-LDP failure does not interrupt multicast traffic forwarding because the mVPN solution does not rely on MPLS.

If a core link fails or is shut down and this link belonged to the path of one or several of the multicast trees, the corresponding traffic (control or data) is affected and some data loss occurs. The traffic is rerouted to a backup path. The delay is mainly because of the IGP convergence time because PIM relies on the unicast routing and triggers a new RPF computation when the IGP topology changes.

Convergence of Default and Data MDTs

Multicast VPN convergence depends on core multicast convergence and MP-BGP convergence characteristics. For in-depth information about multicast, see the convergence subsection in the multicast documentation at the following URL: http://www.cisco.com/en/US/partner/products/sw/iosswrel/ps1829/products_feature_guide09186a00800da1cd.html.

In the case of MP-BGP peer failure or update, connectivity is lost for a single-homed multicast VPN site. Convergence of BGP forces the convergence of the MDTs for sites that depend on more than one PE. See the BGP convergence documents for MP-BGP convergence because it is not different from BGP convergence. Note that the introduction of redundancy for the RRs in the design helps with the availability of MP-BGP.

Convergence of Data MDTs (Establishing a Data MDT)

The data MDTs are dynamically set up after the crossing of a bandwidth threshold by the MDT traffic defined for a given set of sources (ACLs). The statistics to determine whether a multicast stream has exceeded the data MDT threshold are examined once every ten seconds. If multicast distributed switching is configured, the time period can be up to twice as long. The establishment of a data MDT occurs by notification to the PEs of the MD using a message sent to ALL-PIM-ROUTERS (224.0.0.13) on UDP port 3232.

The source PE starts using the data MDT three seconds after sending the notification and also stops using the default MDT. It is possible that some PEs lose some packets of the source multicast flow if they are not already joined to the data MDT at the time the source PE switches over.

Implementing and Configuring the mVPN Service

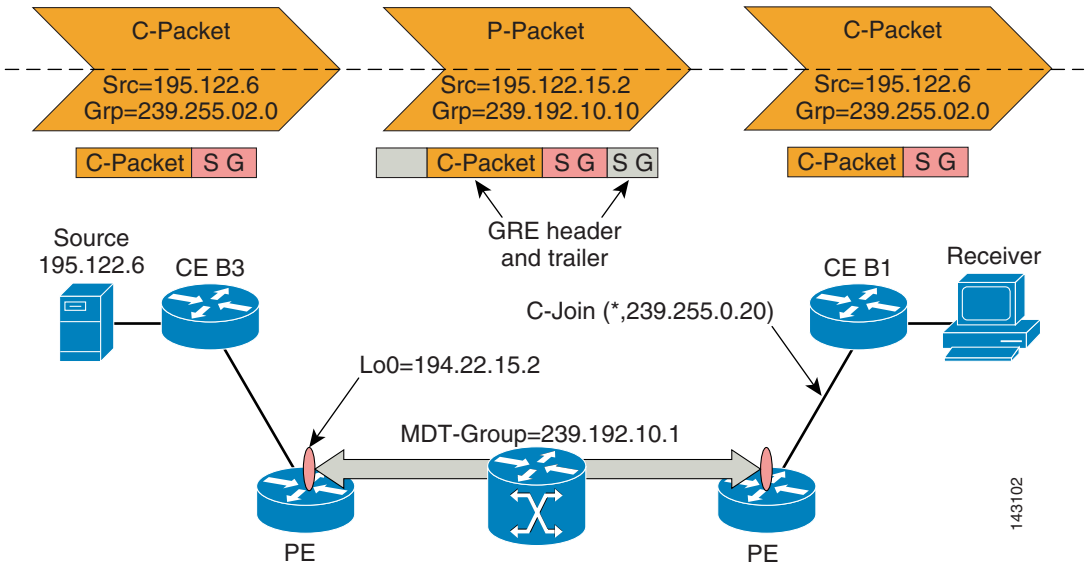
Life of a Packet

Both user control and data traffic (C-packet) is sent over the multicast tunnel. Data traffic for a given multicast group (S,G) or (*, G) is only sent over the multicast tunnel if a remote host registered for the group or if it is a dense-mode user group. The multicast traffic incoming on the PE-CE link is handled as follows:

1. A lookup is done in the MVRF to determine the output interfaces and proceeds to the RPF check. If the tunnel interface is in the outgoing interface list, the packet is encapsulated in GRE (or IP-IP) with the (S, G) information corresponding to S= BGP peering address of the PE router and G = data or default MDT address.
2. The encapsulated multicast packet appears as generated locally by the PE router.
3. A second lookup (including RPF check) is done on this packet in the global multicast routing table. The packet is then forwarded (after appropriate replications) as multicast traffic over the MDT towards the other PE routers. P routers see only P-packets, so they do not build state for traffic inside the VPN. P-packets go to each PE router (except in the case of data MDTs) in the multicast domain. When the P-packet is received from a multicast domain, the PE router duplicates it towards core-facing interfaces if needed. If the MVRF is listed as an entry in the OIF of the global table, it removes the GRE-IP (or IP-IP) encapsulation and then continues to route the packet in the target MVRF identified by the MDT group address in the outer header. The C-packet is sent to the appropriate OIFs according to the MVRF.

Figure 6-14 shows the packet flow including the addressing in the packet headers.

Figure 6-14 Life of a Packet



Configuring the mVPN Service

Use the following commands to enable multicast global and VRF routing:

```
ip multicast-routing [distributed]
ip multicast-routing vrf VRF_name [distributed]
```

Note For cases where MDS is available, use the **distributed** keyword.

Use the following commands to configure the VRF:

```
ip vrf VRF_name
 rd rd
 route-target export rt
 route-target import rt
 mdt default Multicast_Default_MDT_IP@ mdt data <Multicast_Data_MDT_IP@> <range> threshold
threshold list access_list_number
```

Use the following commands to configure the loopback interface for exchanging MP-BGP information:

```
interface Loopback0
 ip address IP_address mask
ip pim sparse-mode
```

All other VRF-linked or core-facing interfaces must be PIM-enabled (sparse-mode or sparse-dense mode, depending on the design options used). Use the following commands to configure PIM on these interfaces: **interface** int_type slot/port

```
ip address address mask ip pim sparse- mode
```

Use the following commands to configure the core multicast, based on the core multicast protocol options:

```
ip pim rp-address IP_address
access_list_number ip pim spt-threshold {infinity | 0 |value}
```

Use the following command to configure SSM in the SSM default range in the core:

```
ip pim ssm range acl-number
           access-list number permit address range
```

Use the following commands to configure the multicast protocol options in the MVRF:

```
ip pim vrf VRF_name rp-address IP_address> access_list_number
ip pim vrf VRF_name bsr-candidate slot/port
ip pim vrf VRF_name rp-candidate slot/port group-list access_list_number
ip pim vrf <VRF_name> ssm range access_list_number
```

For the Cisco 12000, an additional step must be used to enable mVPN:

```
router bgp 1
<snip>
 address-family ipv4 mdt
 neighbor 125.1.125.15 activate
 neighbor 125.1.125.15 send-community extended
 neighbor 125.1.125.16 activate
 neighbor 125.1.125.16 send-community extended
 exit-address-family
```

Use the following commands to enable SNMP and the available MIB notifications:

```
logging source-interface management-interface-slot/port #
syslog server setting logging syslog-server-IP-address
snmp-server community private RW
snmp-server community public RO
snmp-server trap-source management-interface-slot/port
snmp-server location location
snmp-server contact name_or_email
snmp-server enable traps pim neighbor-change rp-mapping-change invalid-pim-message
snmp-server enable traps ipmulticast
snmp-server enable traps msdp
snmp-server enable traps mvpn
snmp-server host SNMP-server-IP-address public
```

For more detailed information, see the following URL:
http://www.cisco.com/en/US/partner/products/sw/iosswrel/ps1839/products_feature_guide09186a0080110be0.html#wp1025098.

Troubleshooting Tips for mVPN

The following are some recommended troubleshooting guidelines:

- Verify the global domain information as follows:
  - Confirm that IP unicast works properly in global domain (IGP and BGP) (multicast uses IP unicast routing information and do not work unless the unicast works correctly).
  - Check PIM neighborships in global domain (PEs-Ps and Ps-Ps).
  - If PIM-SM used, check that an RP is known by all PEs and Ps routers and consistent group—RP mapping exist.
  - Check the mroutes for the default MDT groups on all routers in a path starting with a receiver (make sure to pay attention to flags—check to ensure that the groups shows the correct PIM mode)
- Check the VRF domain information as follows:

- Check unicast VRF tables and neighborships (addresses attached to VRF, route distributions, RTs, RDs, and, if needed, check RR function)
- Check PIM neighborships within VRF (between PEs)
- If PIM-SM used, check that an RP is known by all PE and CE routers consistently (on the VPN user side)
- Check MDT usage (use the **sh ip pim mdt bgp** command)
- Check mroutes on all routers for a path starting with the receiver (make sure group is in correct PIM mode)
- Check traffic flow hop-by-hop

Note Check for the Z flag for default MDT and the Y flag for data MDTs.

Best Practices for Configuring and Troubleshooting mVPNs

The following are some best practices for configuring or troubleshooting mVPNs:

- The update source interface for the Border Gateway Protocol (BGP) peerings must be the same for all BGP peerings configured on the router for the default MDT to be configured properly. If you use a loopback address for BGP peering, then PIM sparse-mode must be enabled on the loopback address.
- The **ip mroute-cache** command must be enabled on the loopback interface used as the BGP peering interface for distributed multicast switching to function on the platforms that support it. Do not enable the **no ip mroute-cache** command on these interfaces.

Useful Show Commands

Table 6-2 lists CLI commands that can be used for troubleshooting mVPN service configuration.

Table 6-2 Useful Show Commands for Troubleshooting mVPN Configuration

Show command	Function
`show ver`	Cisco IOS version—check that mVPN is supported.
`show ip mroute show in`	Interface type of IIF and OIF (Layer 1 and Layer 2; also specify whether sub-interface is used and which type)
`show ip pim mdt`	Number of MVRFs
`show ip pim mdt bgp`	Other PEs multicast-enabled neighbors per MVRF
`show ip pim vrf X neighbor`	Total number of PIM neighbors in each VRF
`show ip pim neighbor"`	Total number of PIM neighbors in global table
`show ip igmp vrf X groups`	Total number of IGMP groups in each VRF
`show ip igmp groups`	Total number of IGMP groups in global domain
`show ip traffic`	Number of PIM/IGMP messages sent/received (itemize by message type)
`show ip mroute vrf x count`	Number of G per MVRF
`show ip mroute vrf x count`	Number of S per G per MVRF
`show ip mroute vrf x`	Number of OIFs per mroute in VRF

Table 6-2 Useful Show Commands for Troubleshooting mVPN Configuration (continued)

`show ip mroute vrf x count`	Memory used for each VRF MRT
`show ip mroute count`	Number of G in global MRT
`show ip mroute count`	Number of S per G in global MRT
`show ip mroute`	Number of OIFs per mroute in global MRT
`show ip mroute count`	Memory used by global MRT
`show mem summary or show proc mem`	Total memory used
`show ip pim vrf x mdt send`	Number of data MDT's sourced
`show ip pim vrf x mdt receive`	Number of data MDT's received
`show proc cpu | exc 0.00`	CPU utilization on RP and LCs
`show ip vrf interfaces` *`vrf_x`*	Interfaces attached to a VRF- Number of CEs per PE per
`show ip vrf` *`vrf_x`*	VRF configuration
`show ip pim mdt bgp`	Visualize the multicast VPN BGP updates
`show run | incl pim show run | incl multicast`	Visualize interface and global PIM and multicast CLIs from the running configuration
`show ip mds interface [vrf` *`vrf-name`*`]`	Display Multicast Distributed Switching (MDS) information for all the interfaces on the line card
`sh ip pim vrf X mdt {send | receive}`	Check the data MDT creation and history
`sh ip mds forwarding`	Check the MDS forwarding information

Ethernet over MPLS

There are requirements within the enterprise network to be able to extend Layer 2 network functionality across a MPLS core to support services such as server clustering, which rely on the ability to use a Layer 2 path between servers, as well as other legacy protocols. For point-to-point Layer 2 connectivity, Ethernet over MPLS (EoMPLS) based on Martini drafts provides this capability.

This section describes how to add EoMPLS as an additional service to MPLS Layer 3 VPNs. The following major sections are included:

- EoMPLS Overview
- EoMPLS Architecture
- Technical Requirements for EoMPLS
- EoMPLS Configuration and Monitoring

EoMPLS Overview

EoMPLS technology leverages an MPLS backbone network to deliver Transparent LAN Services (TLS) based on Ethernet connectivity to the customer site. The concept of Transparent LAN Services is straightforward; it is the ability to connect two geographically-separate Ethernet networks and have the two networks appear as a single logical Ethernet or VLAN domain. Such a VLAN transport capability

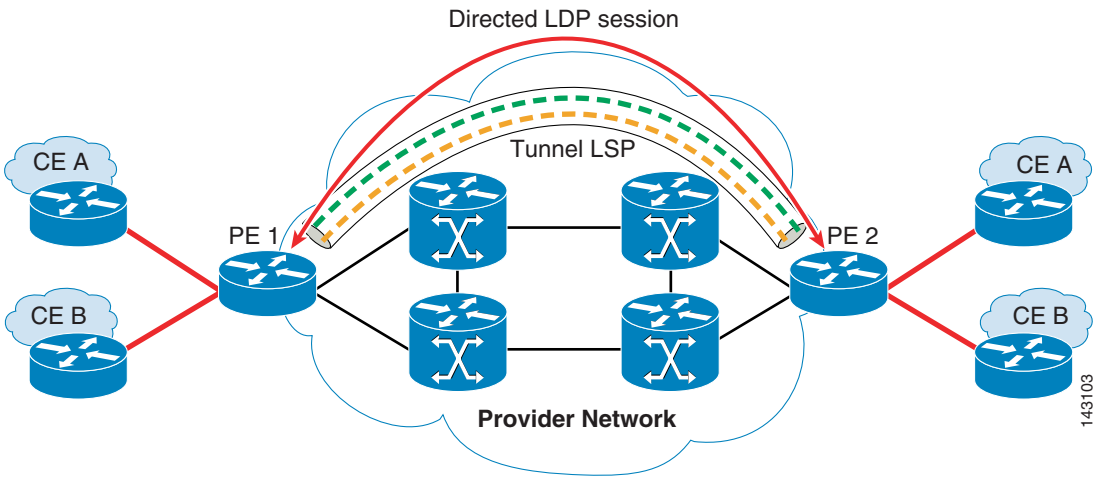
allows customers to deliver a service that allows VLAN networks in different locations within a metro service area to be cost-effectively connected at high transmission speeds, allowing for the ability to carry legacy protocols and provide for Layer 2-based services such as server clustering.

When EoMPLS is deployed in conjunction with MPLS VPN, the network is able to provide tremendous flexibility in the variety of both Layer 2 and Layer 3 network services that can be provisioned over a single, simplified, integrated MPLS backbone network.

EoMPLS Architecture

EoMPLS is a specific implementation of AToM for Ethernet. Figure 6-15 shows the use of a tunnel through an MPLS core network to provide a Layer 2 point-to-point circuit, using AToM, to provide a Layer 2 connection between two separate networks.

Figure 6-15 EoMPLS Architecture



EoMPLS imposition routers must be able to route generic VLAN packets over MPLS backbone. There are four major parts needed to provide EoMPLS service:

- Dynamic MPLS tunnels
- Targeted LDP sessions
- Two-level labeling
- Label imposition/disposition

The EoMPLS implementation is based on the Martini drafts.

A dynamic label tunnel interface is created between the two EoMPLS imposition routers (EoMPLS PEs), an LDP session is formed over this tunnel and MPLS labels are exchanged for each VC to be routed over this tunnel.

Two-level labeling is then used to transport the Layer 2 packets across the backbone. The top label is used to traverse the backbone. The bottom label is the only label seen by the egress LER and is used to identify the VC and the corresponding egress VLAN Layer 2 interface.

EoMPLS has several characteristics that the customer needs to understand to make effective use of this technology:

- Establishing an EoMPLS circuit requires that the edge network be assigned a specific physical port on an LER device, such as a Cisco 7600. The identification of that physical port is a critical element in the binding of the MPLS Label assigned to the customers EoMPLS VC.
- A customer may have more than one EoMPLS VC per physical port as long as the Ethernet traffic transmitted from the customer site to the PE device can be distinguished as having specific 802.1Q headers for each EoMPLS VC by the PE device.
- EoMPLS VCs are point-to-point transmissions only, as explicitly specified in the IETF Martini draft specifications.
- Traffic sent between the imposition/disposition routers (between LERs) over an EoMPLS VC take the same path across the IP/MPLS backbone. The LSP may change because of routing changes inside the MPLS core network.
- Adding/removing a point-to-point Layer 2 VC requires configuration of the two VC endpoints (at the two LERs).
- Provisioning a VC involves defining an endpoint of a Layer 2 VC at each of the VLAN interfaces at the PE router on the interface that connects to the CE.
- The two LERs at the ingress/egress points of the IP/MPLS backbone (the PE routers) are the only routers with knowledge of the Layer 2 transport VCs. All other LSRs have no table entries for the Layer 2 transport VCs. This means that only the PEs require software with EoMPLS functionality.

MPLS VC Circuit Setup

A virtual circuit is an LSP tunnel between the ingress and egress PE, which consists of two LSPs because a uni-directional LSP is required to transport Layer 2 PDUs in each direction. A two-level label stack, where the Level 1 label is the VC label and the Level 2 label is the VLAN tunnel label, is used to switch packets back and forth between the ingress and egress PE.

The VC label is provided to the ingress PE by the egress PE of a particular LSP to direct traffic to a particular egress interface on the egress PE. A VC label is assigned by the egress PE during the VC setup and represents the binding between the egress interface and a given VC ID. A VC is identified by a unique and configurable VC ID that is recognized by both the ingress and egress PE. During a VC setup, the ingress and egress PE exchange VC label bindings for the specified VC ID. The VC setup procedures are transport-independent. The detailed VC setup procedure occurs as follows:

1. An MPLS Layer 2 transport route is entered on the ingress interface on PE1.
2. PE1 starts a remote LDP session with PE2 if none already exists. Both PEs receive LDP KeepAlive messages from each other, reach OPERATIONAL state, and are ready to exchange label bindings.
3. The physical layer of the ingress interface on PE1 comes up. PE1 realizes there is a VC configured for the ingress interface over which Layer 2 PDUs received from CE1 are forwarded, so it allocates a local VC label and binds it to VC ID configured under the ingress interface.
4. PE1 encodes this binding with the VC label TLV and VC FEC TLV and sends it to PE2 in a Label-Mapping message.
5. PE1 receives a Label-Mapping message from PE2 with a VC FEC TLV and VC label TLV. In the VC FEC TLV, the VC ID has a match with a locally configured VC ID. The VC label encoded in the VC label TLV is the outgoing VC label that PE1 is going to use when forwarding Layer 2 PDUs to PE2 for that particular VC.
6. PE1 might receive a Label-Request message from some other PE with a specific VC FEC TLV at any time during the OPERATIONAL state. PE1 examines the VC ID encoded in the FEC element, and responds to the peer a Label-Mapping with the local VC label corresponding to the VC ID.

7. PE2 performs the same Steps 1–6 as PE1. After both exchange the VC labels for a particular VC ID, the VC with that VC ID is fully established.
8. When one LSP of a VC is taken down for some reason, for example, the CE-PE link goes down or the VC configuration is removed from one PE router, the PE router must send a Label-Withdraw message to its remote LDP peer to withdraw the VC label it previously advertised.

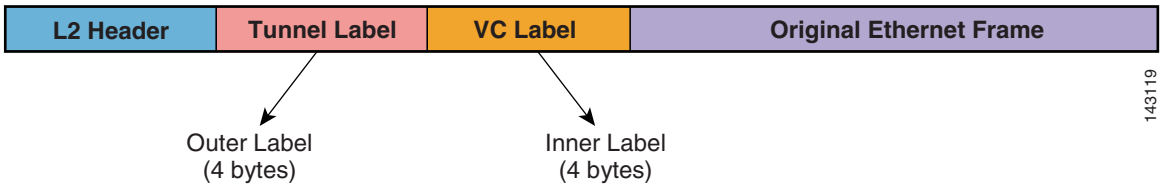
The VC is defined on the inner-most label and is used to bind to the interface to which the packet is to be delivered. The tunnel label is used to forward the packet through the network. Outer labels are assigned to this resultant frame at the egress PE interface for switching trough the MPLS core.

There are two VC types for EoMPLS: type 4 and type 5. Type 4 is used for extending VLANs over MPLS, while type 5 is useful for Ethernet port to port tunneling (port transparency).

Type 5 VCs are used for port-to-port transport of the entire frame without preamble or FCS. BPDUs are also carried transparently across the tunnel.

Only dot1Q vlan tagging is supported.

Figure 6-16 EoMPLS Header



Technical Requirements for EoMPLS

For the Cisco 7600 to perform as an LER to support transport of Layer 2 VLAN packets over MPLS, it must be able to forward 802.1QVLAN Layer 2 VCs across an IP/MPLS backbone. The features required to provide this capability are:

- Targeted LDP session between the peers
- Virtual Circuit between peers
- Two-level labeling
- Label imposition/disposition
- CoS mapping

LDP Session

The ingress PE needs a tunnel label from its upstream IGP neighbor to route the packets it receives from the ingress interface of the ingress PE to the egress PE. The ingress PE and its IGP neighbor are local label distribution peers and maintain a LDP session over which tunnel labels are distributed. They must all have the FEC that contains the host route for the egress PE and the tunnel label is bound to that FEC.

In addition, one targeted LDP session is required between the ingress PE and egress PE, which are remote label distribution peers, to exchange VC labels. All VC label bindings exchanged over this LDP session use the VC FEC element type 128 via the LDP “downstream unsolicited mode.” Because only a single LDP session is required, one is created only if not already present. A session may already be active because another application has already established a targeted LDP session between the two PE routers. LDP, which is only a transport protocol, shall determine that a LDP message is for the AToM application by checking for the existence of the VC FEC element type 128.

Support for the VC FEC element type 128 is required in the following LDP messages:

- Label Mapping Request
- Label Mapping
- Label Mapping Withdraw

If using conservative label retention, the ingress PE also needs to send Label-Request messages for all locally-configured VCs.

If using liberal label retention, the ingress PE does not need to send Label-Request messages for configured VCs, but must prepare to respond to Label-Request sent by other PEs which use conservative label retention. AToM is going to migrate to using liberal label retention mode, but initially the implementation employs conservative label retention mode.

Two Level Labeling

The Layer 2 transport service over MPLS is implemented through the use of two level label switching between the edge routers. The label used to route the packet over the MPLS backbone to the destination PE is called the "tunnel label." The label used to determine the egress interface is referred to as the VC label. The egress PE allocates a VC label and binds the Layer 2 egress interface to the VC in question, then it signals this label to the ingress PE via the targeted LDP session.

Label Imposition/Disposition

When the Layer 2 PDU arrives at the ingress interface of the ingress PE, the router must perform label imposition and switch the packet to the appropriate outgoing MPLS interface which routes the packet to the egress LER for the VC in question.

When the egress PE receives the packet, it receives the packet with only the VC label because its neighbor (known as the penultimate router) pops the tunnel label before forwarding the packet. The egress PE uses the VC label to perform disposition and switch the packet to the appropriate egress interface.

Class of Service (QoS) Mapping

MPLS provides QoS using the three experimental bits in a label to determine the queue of packets. To support QoS from PE to PE, the experimental bits in both the VC and Tunnel labels must be set. The experimental bits need to be set in the VC label because the tunnel label is popped at the penultimate router. In the case of EoMPLS, two methods of setting experimental bits are provided.

Static Setting of Experimental Bits

The customer is able to configure the PE to set the EXP bits in the labels to a given value based in ingress interface.

Using VLAN User Priority Bits to Determine Experimental Bit Settings

The three user priority bits are used to index into a table of eight values. The value for a given index is used to set the experimental bits. This method may cause out-of-order packets when packets have different user priorities.

Load Sharing—Ingress PE

Even there are multiple equal-cost routes between the two LERs, there is no load sharing done on ingress LER, because packets still have an Layer 2 header when EARL does the FIB lookup.

Load Sharing—MPLS Core

When EoMPLS packets get into the core of MPLS, the load sharing behavior may be different depending on how LSR looks into the packets. Assuming LSRs are also Constellation 2 systems, and then with control word inserted, EoMPLS packet flows are load shared among the equal paths because by checking the first nibble that is below the label stack, earl7 parses down as deep as the fifth or lowest label to use for hashing. Without the control word inserted, EoMPLS packet flows behavior over equal cost paths are different depends on the value of MAC address under the label stack. That is the reason why Constellation 2 and Cisco IOS inserts control word to EoMPLS packets by default.

Currently, the Cisco 7600 supports two main models of EoMPLS:

- PFC-based EoMPLS (No Local Switching)

  In this mode, the PFC3B/XL performs all label imposition /deposition functionality, as well as any required Ethernet encapsulation and 802.1Q header insertion.

  VC type is a result of negotiation between the two PEs, type 4 and 5 are both supported.

  IP ToS is always preserved end-end.

- PXF-based EoMPLS (With Local Switching)

  The egress interface is responsible for all label imposition and deposition, resulting in the requirement that this be an OSM interface on egress. The PFC simply forwards any packet received that contains a MAC address from the other side to the egress interface. The PFC does not take into consideration the EoMPLS overhead.

  For PXF-based EoMPLS, OSM cards are required for the core-facing interface.

  VC type is a result of negotiation between the two PEs; type 4 and 5 are both supported.

  This mode supports local switching, which may be useful for providing access-level redundancy.

  If ingress port is marked as Trust DSCP, CoS is not preserved.

  When using **mls qos queueing-only**, ingress PFC CoS and DSCP are preserved because you bypass the PFC QoS mechanisms; shaping and queuing are done on the egress OSM interface.

For CoS preservation, Type 5 VCs result in the stripping of the dot1Q header at the ingress PE, so there is no preservation. If at the egress PE, **mls qos** is enabled, the EXP bits are mapped back to the CoS bits. On ingress ports, use the command **mls qos trust cos**.

EoMPLS Restrictions

Dynamic IP Labeling

To support this feature, Dynamic IP labeling (**mpls ip**) must be enabled an all paths between the two imposition/disposition LERs. Failure to do so results in the packet being discarded before it reaches the disposition PE.

Summarized Routes

Routes from a PE discovered by its peers must be unsummarized; that is, address/32. This is required to ensure that there is an LSP from PE to PE.

Frame Size

The MPLS network should be configured with an MTU that is at least of 12 bytes plus link header larger than the largest frame size that is transported in the LSPs.

CE-side MTU or SP core links must be changed to accommodate the encapsulation overhead.

If a packet length, after it has been encapsulated on the ingress LSR, exceeds the LSP MTU, it must be dropped.

If an egress LSR receives a packet on an EoMPLS VC with a length, after the label stack and control word have been popped, that exceeds the MTU of the egress Layer 2 interface (VLAN), it must be dropped.

Fragmentation is not supported for Layer 2 packets transmitted across the MPLS backbone. Therefore to deploy the layer transport service, the network operator must make sure that the MTU of all intermediate links between the endpoints is sufficient to carry the largest Layer 2 transported packets that are received. The MTU setting of the ingress PE needs to match with the setting at the egress PE. A VC does not properly establish if MTU sizes mismatch.

Configuration and Monitoring

For this design guide, all testing on the Cisco 7600 was done with PFC-based EoMPLs, rather than PXF-based.

PXF-Based Cisco 7600 Configuration

```
class-map match-any L2VPN_TRAFFIC
  match any
policy-map SET_L2VPN_TRAFFIC
  class L2VPN_TRAFFIC
   set mpls experimental 3
interface Loopback0
ip address 1.1.1.1 255.255.255.255

interface GigabitEthernet9/1.1
encapsulation dot1Q 2000
xconnect 2.2.2.2 200 encapsulation mpls
service-policy input SET_L2VPN_TRAFFIC
interface GE-WAN9/3 (core facing interface)
ip address 10.10.93.2 255.255.255.0
negotiation auto
tag-switching ip
mls qos trust dscp
```

Cisco 12K Configuration

```
interface Loopback0
ip address 1.1.1.1 255.255.255.255   !Loopback address Must be /32
interface GigabitEthernet9/1
mtu 9216
interface GigabitEthernet9/1.1
encapsulation dot1Q 2000 this can be used
xconnect 2.2.2.2 200 encapsulation mpls    ! Remote Loopback addr and VC id
```

Cisco 7200 Configuration

```
class-map match-any L2VPN_TRAFFIC
  match any
policy-map SET_L2VPN_TRAFFIC
  class L2VPN_TRAFFIC
   set mpls experimental 3
```

```
interface Loopback0
ip address 10.191.44.251 255.255.255.255
interface GigabitEthernet0/2
 mtu 9216
 no ip address
 load-interval 30
 duplex full
 speed 1000
 media-type rj45
 no negotiation auto
!
interface GigabitEthernet0/2.20
 encapsulation dot1Q 20
 no snmp trap link-status
 no cdp enable
 xconnect 10.191.44.252 20  encapsulation mpls
 service-policy input SET_L2VPN_TRAFFIC
```

Cisco 3750 Metro Configuration

```
class-map match-any L2VPN_TRAFFIC
  match any
policy-map SET_L2VPN_TRAFFIC
  class L2VPN_TRAFFIC
   set mpls experimental 3

interface Loopback0
ip address 10.191.44.252 255.255.255.255
interface GigabitEthernet0/2
 mtu 9216
!
interface GigabitEthernet0/2.20
 encapsulation dot1Q 20
 no snmp trap link-status
 no cdp enable
 xconnect 10.191.44.251 20  encapsulation mpls
 service-policy input SET_L2VPN_TRAFFIC
```

Cisco PXF-Based and Cisco 12K Monitoring Commands

```
7600#sh mpls l2transport vc
Local intf Local circuit Dest address VC ID Status
------------ ------------------- --------------- ---------- ----------
Gi9/1.1 Eth VLAN 2000 2.2.2.2 200 UP VC Status

7600#sh mpls l2transport vc detail
Local interface: Gi9/1.1 up, line protocol up, Eth VLAN 2000 up
Destination address: 2.2.2.2, VC ID: 200, VC status: up
Tunnel label: 16, next hop 10.10.93.1
Output interface: GE9/3, imposed label stack {16 2000} Two level label
Create time: 01:37:35, last status change time: 00:00:13
Signaling protocol: LDP, peer 2.2.2.2:0 up
MPLS VC labels: local 21, remote 2000
Group ID: local 0, remote 0
MTU: local 1500, remote 1500
Remote interface description:
Sequencing: receive disabled, send disabled
VC statistics:
packet totals: receive 234343, send 53196336
```

```
byte totals: receive 232323332, send 3191872568
packet drops: receive 0, send 0

7600#remote command switch show mpls l2transport vc detail
Local interface: GigabitEthernet9/1, Eth VLAN 2000
Destination address: 2.2.2.2, VC ID: 200 VC TYPE
VC status: receive UP, send UP
VC type: receive 5, send 5
Tunnel label: 16, next hop 10.10.93.1
Output interface: GE9/3, imposed label stack {16 2000}
MPLS VC label: local 21, remote 2000
Linecard VC statistics:
packet totals: receive: 234343 send: 38222872
byte totals: receive: 232323332 send: 2293372320
packet drops: receive: 0 send: 0

7600#sh mpls l2transport binding
Destination Address: 2.2.2.2, VC ID: 200
Local Label: 21
Cbit: 0, VC Type: Ethernet, GroupID: 0
MTU: 1500, Interface Desc: n/a
Remote Label: 2000
Cbit: 0, VC Type: Ethernet, GroupID: 0
MTU: 1500, Interface Desc: n/a

7600#sh mls cef eom
Index VPN Adjacency
128 257 278528,0
7600#sh mls cef adjacency entry 278528 detail
Index: 278528 smac: 000b.fcd4.cf00, dmac: 00d0.0362.7800
mtu: 1518, vlan: 1020, dindex: 0x0, l3rw_vld: 1
format: MPLS, flags: 0xD000008400 Hardware Programmed MTU
label0: 0, exp: 0, ovr: 0
label1: 2000, exp: 0, ovr: 0
label2: 16, exp: 0, ovr: 0
op: PUSH_LABEL2_LABEL1
packets: 69787014, bytes: 4187220840
```

Cisco PFC-Based Configuration

```
vlan 2000  !configure vlan in global database
interface GigabitEthernet9/1 ! CE facing port as Access/Trunk Port
no ip address
switchport
switchport trunk encapsulation dot1q
switchport mode trunk
interface Vlan2000
no ip address
xconnect 1.1.1.1 200 encapsulation mpls
```

Cisco PXF-Based Monitoring Commands

```
7600#sh mpls l2transport vc
Local intf Local circuit Dest address VC ID Status
------------ -------------------- --------------- ---------- ----------
Vl500 Eth VLAN 500 1.1.1.1 200 UP

7600#sh interfaces trunk
Port Mode Encapsulation Status Native vlan
Gi8/2 on 802.1q trunking 1
Port Vlans allowed on trunk
```

```
Gi8/2 1-4094
Port Vlans allowed and active in management domain
Gi8/2 1,2000
Port Vlans in spanning tree forwarding state and not pruned
Gi8/2 1,2000
7600#ipc-con 8 0    ⇓Remote Login to PXF card Imposing label
Entering CONSOLE for slot 8
Type "^C^C^C" to end this session
CWTLC-Slot8>en
CWTLC-Slot8#sh mpls l2transport vc
Local intf Local circuit Dest address VC ID Receive/Send
VC Status
------------ ------------------ --------------- ---------- ------------
Vl2000 Eth VLAN 500 1.1.1.1 200 UP /UP
CWTLC-Slot8#sh mpls l2transport vc de
Local interface: Vlan2000, Eth VLAN 2000
Destination address: 1.1.1.1, VC ID: 200
VC status: receive UP, send UP
VC type: receive 5, send 5
Tunnel label: 16, next hop 10.10.191.2
Output interface: GE8/1, imposed label stack {16 23}
MPLS VC label: local 21, remote 23
Linecard VC statistics:
packet totals: receive: 0 send: 0
byte totals: receive: 0 send: 0
packet drops: receive: 0 send: 0
Control flags:
receive 1, send: 11
CWTLC EoMPLS disp detailed info: AC if_no 30
t vclbl VLAN Type h impidx stat
- d------- x---(d---) ------- - x-- x---
0 00000021 07D0(2000) ether 1 1C4 0001
1 00000021 07D0(2000) ether 1 1C4 0001
vlan(2000) rx_pkts(0)
CWTLC EoMPLS imp detailed info: AC if_no 30, Egress GE-WAN8/1
Vlan func[0]: 2000(0x7D0): flag(0x0) func(3:atom ether) hash (0x1)
Tx TVC Table:
idx ltl op vcinfo en h next intfid
x--- x-- -- d----- -- - x--- x-----
tx-tvc 0044 004 02 000021 00 1 0000 000000 pxf[0] vlan: 2000 hash:001
TTFIB: Index(21) Imposition(PUSH 2):{16, 23, 0, 0}    ⇓PXF Entries
t VLAN vc lbl tun lbl MAC Address cos2exp
```

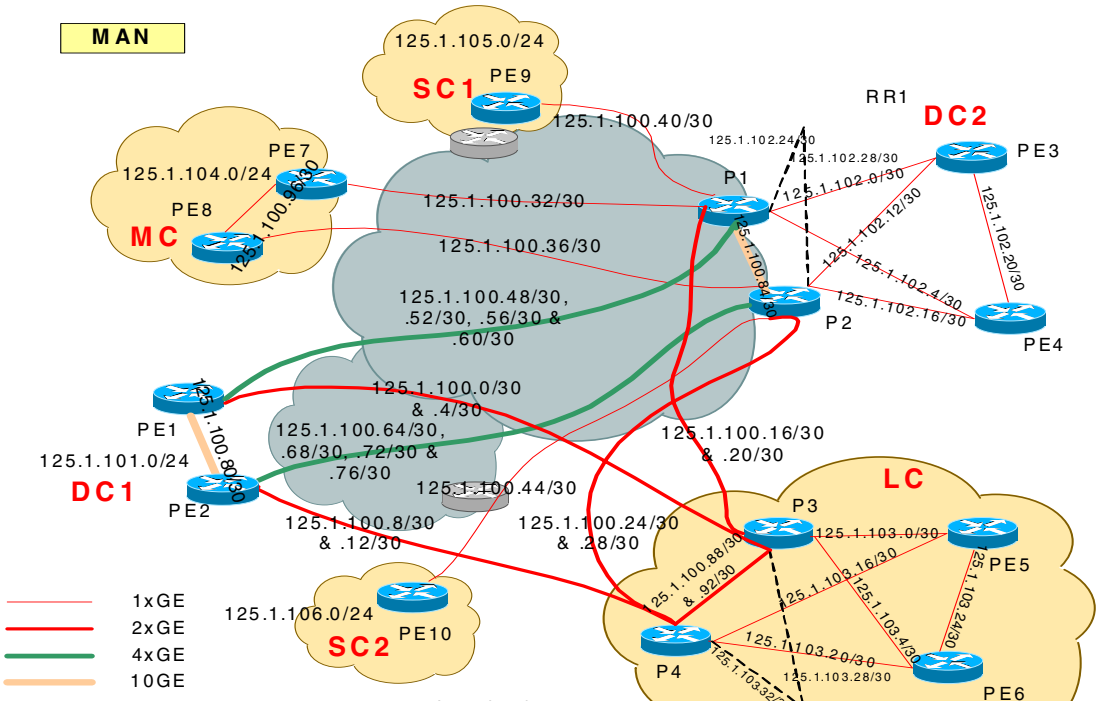
MPLS-Based VPN MAN Testing and Validation

Test Topology

The test topology was designed to encompass the most common elements of a typical MAN network: data centers and different-sized campuses (small, medium, and large). For this phase of the testing, an entire network consists of full-rate GE interfaces.

Figure 7-1 shows the MPLS MAN core (P, PE, and connectivity between them).

Figure 7-1 MPLS MAN Core Topology



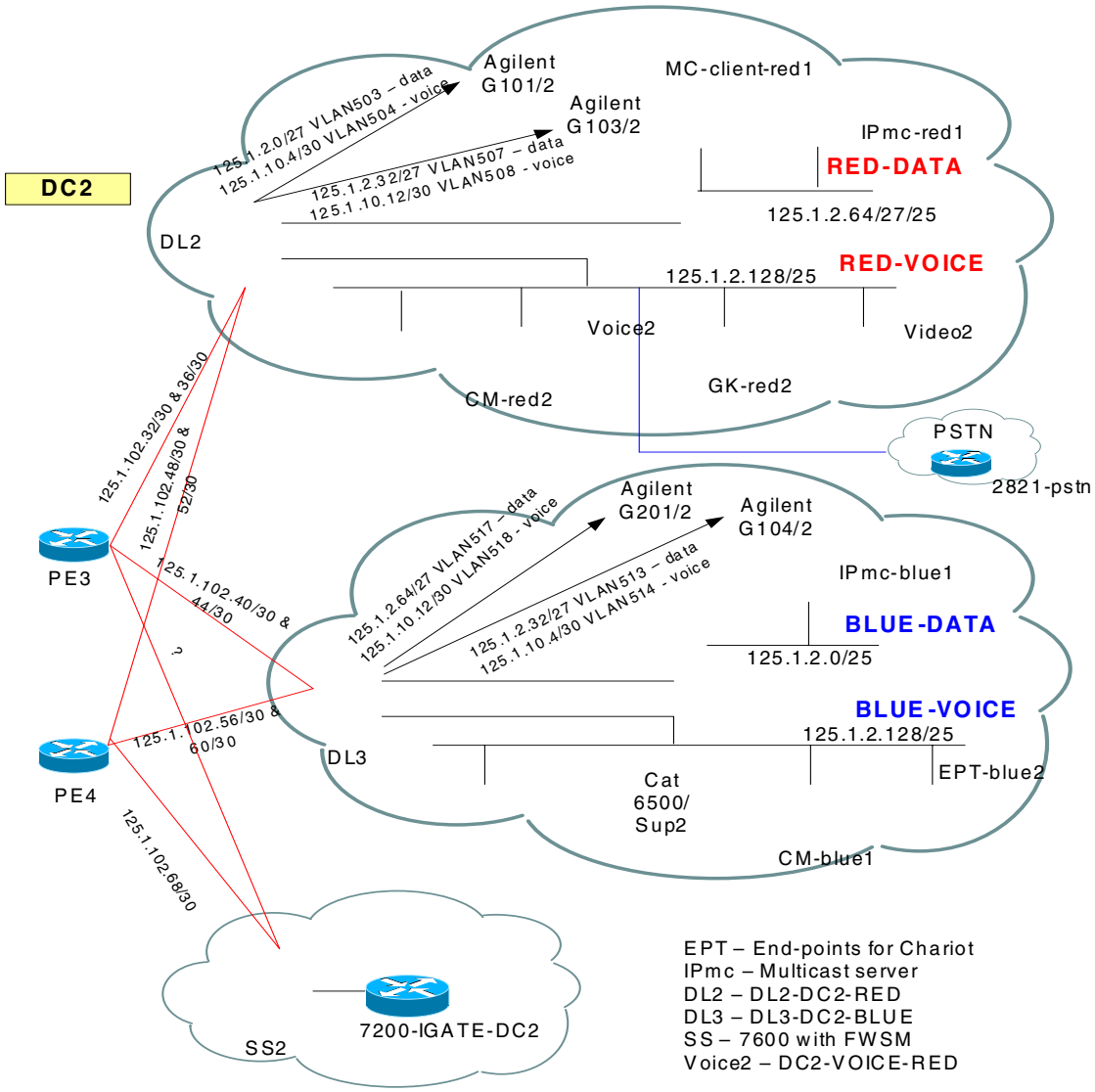
		Loopback
P1	12k-DC2-P1	125.1.125.1
P2	7600-DC2-P2	125.1.125.2
P3	12k-LC-P3	125.1.125.3
P4	7600-LC-P4	125.1.125.4
PE1	7600-DC1-PE1	125.1.125.5
PE2	7600-DC1-PE2	125.1.125.6
PE3	7600-DC2-PE3	125.1.125.7
PE4	12k-DC2-PE4	125.1.125.8
PE5	7600-LC-PE5	125.1.125.9
PE6	7304-LC-PE6	125.1.125.10
PE7	7600-MC-PE7	125.1.125.11
PE8	7600-MC-PE8	125.1.125.12
PE9	7200-SC1-PE9	125.1.125.13
PE10	3750-SC2-PE10	125.1.125.14
RR1	7200-DC2-RR1	125.1.125.15
RR2	7200-LC-RR2	125.1.125.16
SS1	7600-DC1-SS1	125.1.125.17
SS2	7600-DC2-SS2	125.1.125.18

VRF red-data	10:103; 10:103x
VRF red-voice	10:104; 10:104x
VRF blue-data	10:105; 10:105x
VRF blue-voice	10:106; 10:106x
VRF green-data	10:107; 10:107x

x denotes PE number in case of dual homed CEs – configured on higher #ed PE within a cloud. For eg: PE1 would have vrf red-data as 10:103 & PE2 would have 10:1032

Figure 7-2 shows one of the data centers (DC2), which is a representative example of the rest of the network.

Figure 7-2 Representative Data Center (DC2)



Some of the key features of the test bed are:

- Roles based testing—Multiple platforms for various roles (P, PE, RR, and CE, as shown in Table 7-1).
- End-to-end redundancy—Links, processors, functional routers, and so on.
- Individual as well as integrated system-level testing of the features—Features were classified under two broad categories: network services such as QoS, high availability, and security; and enterprise services such as voice, multicast, and data applications (see Table 7-2).

The overall intent of the testing was to validate and demonstrate the working of enterprise services over an MPLS VPN-based MAN and to provide best practices in terms of network services configuration.

Table 7-1 Test Platforms and Roles

Platform	Role	SW Version	LC—Core facing*	LC—Edge facing*	RP
Cisco 12000	P	12.0(31)S	EPA-GE/FE-BBRD EPA-3GE-SX/LH-LC (E4+)	EPA-GE/FE-BBRD EPA-3GE-SX/LH-LC (E4+)	PRP-1
Cisco 12000	PE	12.0(31)S	EPA-GE/FE-BBRD EPA-3GE-SX/LH-LC (E4+)	4GE-SFP-LC (E3)	PRP-2
Cisco 7600	P	12.2(18)SXE2	WS-X6724-SFP	WS-X6724-SFP	Sup720-3BXL
Cisco 7600	PE	12.2(18)SXE2	WS-X6724-SFP	WS-X6724-SFP	Sup72-3BXL
Cisco 7200	PE	12.2(25)S5	Built-in	Built-in	NPE-G1
Catalyst 3750 ME	PE	12.2(25)EY2	Built-in	Built-in	N/A
Cisco 7304	PE	12.2(25)S5	SPA-2GE-7304	SPA-2GE-7304	NSE-100
Catalyst 6500	CE/DL	12.2(18)SXE2/SXE1	6408A/6416	6408A/6416	Sup720/Sup2
Cisco 7200	RR	12.0(30)S1	PA-GE	PA-GE	NPE-400

Table 7-2 Testing Features

Baseline Architecture	Function
MPLS, LDP, IGP, and so on	MPLS as transport
Baseline MPLS-VPN	VPNs for segmentation in the MAN
Network Services	
QoS	Provide QoS profile/classification/implementation guidance for the MAN
Security (FW/NAT/ACLs and so on)	Security for shared services and Internet
HA (TE, FRR, Fast IGP)	
Management	ISC 4.0 for provisioning and management
End-to-End Services	
Voice	Verify that the existing overlay architecture works
Multicast	Demonstrate mVPN implementation in MAN along with its integration with existing multicast architecture
Data center traffic	Verify that the standard traffic profiles/parameters are supportable
Competitive Tests	
M10i	System tested as a PE

The following are additional details of the test topology:

- Segmentation was assumed to based on traffic type. Thus there were two organizations (red and blue) with both data and voice VRF, and one organization (green) with data-only VRF.
- For every organization, voice and data had a separate VLAN.
- In the case of red and blue organizations, the distribution layer (DL) switches were configured with VRF-lite to extend the traffic separation into the campus/data center. The green DLs did not have VRF-lite.
- Red was configured to use OSPF for PE-CE protocol, while blue and green were set up to use EIGRP.
- Shared services gateways were set up that could be configured in redundant, non-redundant, or load balancing modes, depending on the application(s) being shared.
- Out-of-band access was provided for every device in the MAN for management purposes.

Test Plan

Although all the services were tested as a system, they each had a focused test plan as well. This section discusses some of the high level details.

Note The focus of the testing was not to test scale/performance of the services but to validate them as an end-to-end, system-level proof-of-concept.

Baseline MPLS VPN

The baseline MPLS VPN was set up to demonstrate:

- IGP variations in the MAN core—Usage and differences in EIGRP versus OSPF as the core IGP
- IGP variations at the MAN edge—Usage and differences in EIGRP versus OSPF as the core IGP
- Multipath configuration and implications within the VPN for VRF routes
- Multipath configuration and implications within the MAN core for PE-to-PE reachability routes
- Route reflector-based MP-iBGP meshing (including redundancy)
- End-to-end traffic convergence with and without tuning (IGP and BGP)
- Cisco Express Forwarding load-balancing in situations with multiple back-to-back links

Security

The purpose of the security testing was to demonstrate the integration of MPLS VPNs and shared services and access to these services through virtual firewalls. Security testing in the MPLS MAN focused on the following areas:

- Common services area routing—Routing mechanism that allows the VPNs to communicate with the common services area and among themselves.
- Dual Internet access—Redundant Internet access was provided to optimize the use of the MAN, which was achieved in the two following ways:

  - Equal cost MAN—Injects two equal cost default routes into the MAN with the routing protocol choosing one route over another, depending on the distance to the exit point in the MAN.
  - Engineered exit points—Leverages the capabilities of MP-iBGP to selectively exchange default routes between VRFs to engineer the exit point based on the location.
- Centralized Services: Centralized services were divided into two kinds:
  - Shared services:
    - Protected services—Protected services are to be accessed through the firewalls connected to the shared services routers.
    - Unprotected services—Access to non-firewalled segments by route imports/exports.
  - Dedicated (per-VPN) Services: Services such as DHCP were deployed separately for each VPN.

QoS

Because this phase used full-rate GE within a wholly-owned enterprise MAN, only queueing was implemented. Other requirements such as shaping (in the case of sub-rate GE) will be addressed in future phases.

End-to-end QoS was implemented (CE-to-CE) with the following objectives:

- Testing whether the 8-class model (11 classes within the data center/campus mapped to 8 classes within MAN) can be maintained within the MAN (especially on GSR).
- Ensuring that the traffic is queued according to the configured trust parameter (dscp, cos, exp) at each egress queue.
- Ensuring that priority traffic such as real-time gets prioritized over other classes of traffic in case of congestion without affecting delay or jitter.
- Testing QoS characteristics with real applications (voice calls, multicast server/clients) rather than to simply test tools such as Agilent.

Overall, the network had a large amount of redundancy/load-sharing built-in using high bandwidth links; thus the PE-CE link was considered the ideal test point for creating bottlenecks. Other than basic validation in the core, the focus was on the PE-CE link across various PE platforms.

Data

For this phase of testing, the real value-add did not require simulating actual data applications and thus a test tool such as Agilent was considered sufficient. Agilent was configured to generate traffic flows (multiple source/destination pairs; typically 254x254 flows for each class of service) for both data as well as voice VRFs. The traffic flows were separated based on the different classes of service that were expected to be seen in the core and the correct DSCP values were set. Although the major focus of the testing was on voice, data traffic was ideal for measuring packet losses and for varying the link usage rates.

Voice

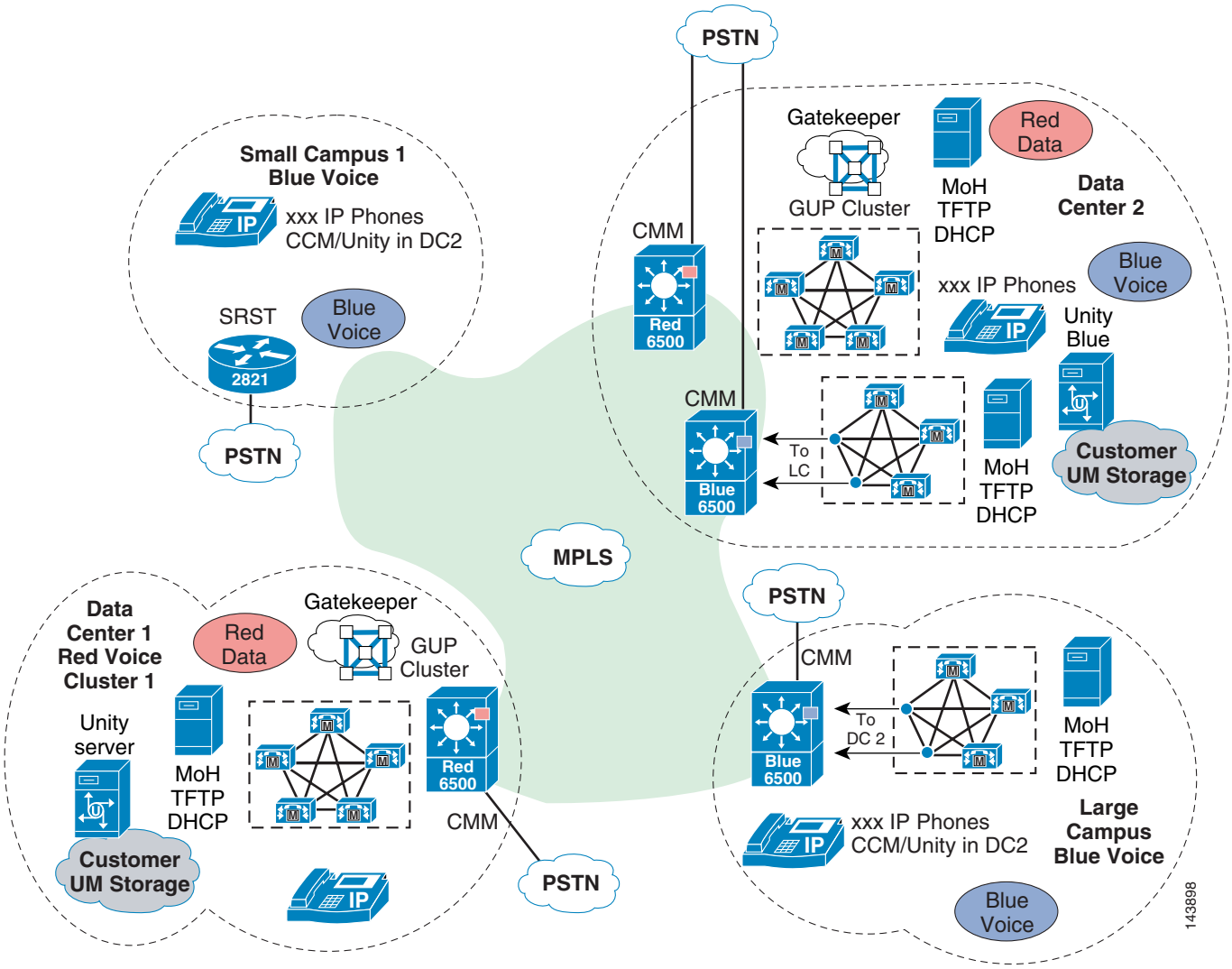
Phase 1.0 tested the following three deployment models:

- Clustering over the MAN

- Centralized call processing with SRST
- Multi-site distributed call processing

In addition, solution test areas included end-to-end functionality, local and centralized Cisco Unity messaging, stress, load, QoS, redundancy, reliability, usability, availability, music on hold, fax, endpoint configuration, and TFTP downloads for upgrades. The intention of stress testing the network was not to validate the limits on individual components of the solution but to validate that the solution remains stable over an extended period of time while subjected to call scenarios expected in real-life deployments. Figure 7-3 shows an overall view of the voice components.

Figure 7-3 Voice Components



Following is a listing of the major products and features tested:

- CCM 4.1(3)sr1
- Unity 4.0(4)
- Catalyst CMM and Cisco IOS-based gateways
- Conferencing

- VG248 gateway
- SRST
- Fax relay, fax pass-through
- QoS
- Connection Admission Control (CAC)
- DHCP configuration
- TFTP
- Multicast music on hold
- Extension mobility

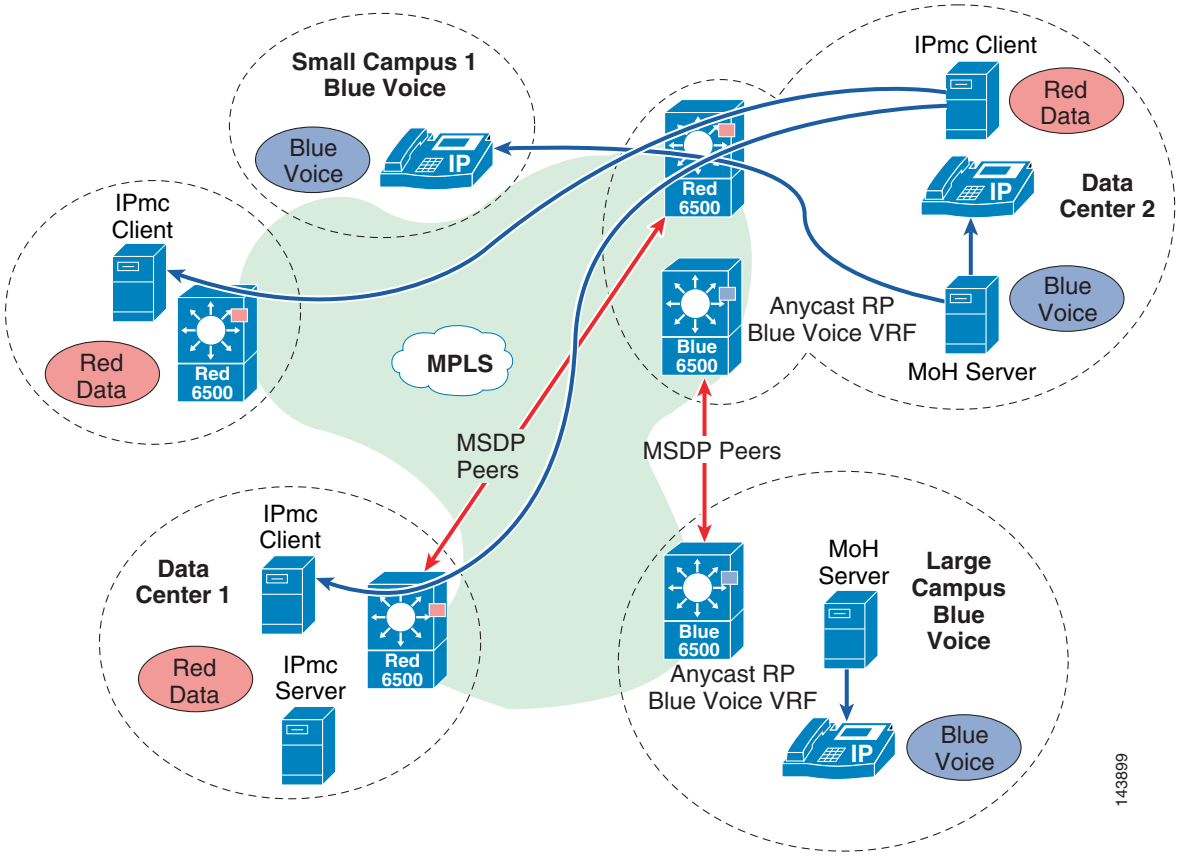
Because of equipment and time constraints, Phase 1.0 testing did not include following features (which are targets for Phase 2 testing):

- Contact Center applications
- IP Communicator
- VT Advantage and video conferencing
- Attendant console
- Inter-VPN voice connectivity

Multicast

Figure 7-4 shows the multicast test setup.

Figure 7-4 Multicast Test Setup



The multicast test setup had the following highlights:

- Anycast RP was used to ensure that the closest RP was selected within the network.
- The RPs were peered using MSDP within each VRF.
- mVPN was used within the MPLS network to natively forward the multicast traffic (non-MPLS switched).
- Agilent behaved as sender/receiver of multicast traffic, enabling the creation of multiple streams.
- An actual streaming video multicast from a server to multiple clients.
- The clients were set up locally within the same Layer 2 network as the server as well as across the MPLS network to visually compare any degradation that may occur across the network across various traffic rates.

MPLS Network Convergence

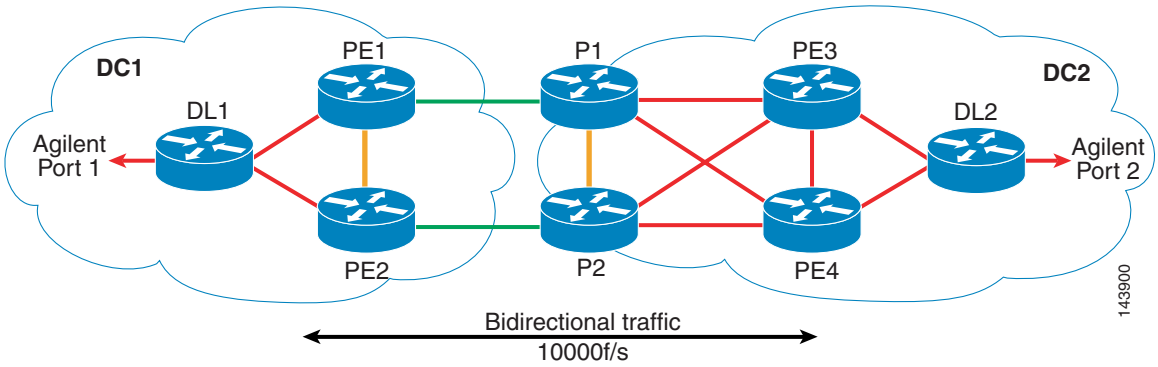
Except for possibly multicast, none of the other applications should have a dependency on the MPLS network. Thus while the application architectures may not change much, their dependency on two critical network needs becomes even more important: end-to-end QoS treatment and network convergence.

As already discussed, QoS focused on CoS, queuing, and prioritization. Convergence testing focused on ensuring end-to-end convergence of the traffic with minimal/no packet loss. Both EIGRP and OSPF were used as the core IGP and PE-CE protocols. The tests were done to measure both up and down convergence and were performed in two phases: an untuned network (default timers) and a tuned network (BGP and IGP timers tuned for faster convergence). The purpose was to present two extreme proof points based on which a customer network can be tuned to achieve the desired level of convergence.

Convergence Test Results

The network topology shown in Figure 7-5 was used to test the various failure scenarios.

Figure 7-5 Convergence Test Topology



Note For all the test results, A/B means traffic received on Port1/Port2 of Agilent.

The purpose of the convergence tests was to capture end-to-end traffic convergence in case of various failure scenarios. The failure scenarios were:

- PE-CE link failure (PE3-DL2)
- P-PE link failure (P1-PE3)
- PE failure (PE3)
- P failure (P1)

The tests were conducted by shut/no shut of links and reloading the appropriate routers depending on the test. Three iterations were done for each failure scenario and the max bidirectional traffic times reported. Any packet loss greater then 5% of the traffic sent was recorded. The tests were also classified based on the core IGP protocol being used—EIGRP or OSPF; PE-CE protocol being used—EIGRP or OSPF; and Routing protocol state—untuned or tuned.

The tuning configuration template used for each of the protocols was:

- BGP—The focus was to tweak the import scanner, the nexthop check scanner and the advertisment intreval on the PEs as well as RR.

```
router bgp 1
  bgp scan-time 5
 address-family vpnv4
  bgp scan-time import 5
  bgp scan-time 5
  neighbor 125.1.125.15 advertisement-interval 1
```

```
neighbor 125.1.125.16 advertisement-interval 1
```

Next hop event based tracking feature is not currently supported on Cisco 7600.

- OSPF—The focus was to improve the dead neighbor detection, LSA generation, and the SPT calcuations.

```
interface Gx/y
carrier-delay msec 0

router ospf 1 vrf red-data
timers throttle lsa all 0 20 5000
timers lsa arrival 20
timers pacing flood 8
timers throttle spf 24 24 5000
```

Optionally if you have a multipoint/broadcast interface, sub-second hellos can be enabled by the command **ip ospf dead-interval minimal hello-multiplier** <3-20>. This sets the number of hellos sent per second. The dead intreval is always 4 times the hello intreval.

- EIGRP—No tuning is required.

Note MPLS VPN NSF/SSO is not currently supported on 12xxx or 7600 and hence was not tested. TE-FRR will be tested in the next phase of the solution as well.

The results of these tests are presented below. Within each iteration, the x/y format represents the bidirectional traffic convergence numbers—x for traffic from port2 to port1 (right to left in the figure) and y for traffic from port1 to port2 of agilent (left to right in the figure).

Some observations:

- Because multipath was turned on, any packet loss observed was restricted to the traffic flowing the failure link or router.
- Overall the up convergence (routers/links coming up) reported no packet losses in most scenarios.
- In most failover scenarios even with the tuned protocols, the BGP import scanner is the deciding factor (minimum of 5 sec).
- Since all the failures were emulated in DC1, in most cases traffic y took the largest amount of time. For example, in case of PE3 failure, traffic from PE1 had to reconverge and forwarded via PE2 since no other direct link to DC2 existed.
- P down convergence was much faster with OSPF then EIGRP.
- Overall P failure demonstrated the highest traffic times and percentage of packet losses. An additional link between PE1 and P2 would have most likely helped improve the convergence times.

Core IGP—EIGRP

Table 7-3 PE-CE Protocol—OSPF Untuned

Scenario		T1 (s) (port1/port2)	T2 (s) (port1/port2)	T3 (s) (port1/port2)	Max time for bidirectional conv (s)	Max % packet loss (port1/port2)
PE3-DL2 failure						
	down	2/18	2/18	2/18	18	25%/50%

Table 7-3 PE-CE Protocol—OSPF Untuned

	up	0	0	0	0	0%
P1-PE3 failure						
	down	9/2	6/2	5/3	9	25%/10%
	up	0/2	0/2	0/25ms	2	0%/5%
PE3 failure						
	down	3/10	3/9	4/8	10	10%/50%
	up	0	0	0	0	0%
P1 failure						
	down	19/19	19/22	20/23	23	50%/40%
	up	34/32	28/29	5/9	34	30%/30%

Table 7-4 PE-CE Protocol—OSPF Tuned

Scenario		T1 (s) (port1/port2)	T2 (s) (port1/port2)	T3 (s) (port1/port2)	Max time for bidirectional conv (s)	Max % packet loss (port1/port2)
PE3-DL2 failure						
	down	2/9	2/9	2/9	9	20%/50%
	up	0	0	0	0	0%
P1-PE3 failure						
	down	6/3	6/3	4/2	6	25%/10%
	up	0	0	0	0	0%
PE3 failure						
	down	0/5	0/7	0/6	7	0%/35%
	up	0	0	0	0	0%
P1 failure						
	down	17/23	20/22	18/21	23	50%/40%
	up	5/7	2/8	900ms/10	10	25%/25%

Table 7-5 PE-CE Protocol—EIGRP Untuned

Scenario		T1 (s) (port1/port2)	T2 (s) (port1/port2)	T3 (s) (port1/port2)	Max time for bidirectional conv (s)	Max % packet loss (port1/port2)
PE3-DL2 failure						
	down	2/11	3/12	2/12	12	30%/50%
	up	0	0	0	0	0%
P1-PE3 failure						
	down	5/3	5/2	4/3	5	25%/10%

Table 7-5 PE-CE Protocol—EIGRP Untuned

	up	0/2	0	0	2	0/5%
PE3 failure						
	down	3/6	2/4	3/5	6	15%/35%
	up	0	0	0	0	0%
P1 failure						
	down	18/22	16/23	10/21	23	15%/50%
	up	5/8	0/13	0/9	13	5%/25%

Table 7-6 PE-CE Protocol—EIGRP Tuned

Scenario		T1 (s) (port1/port2)	T2 (s) (port1/port2)	T3 (s) (port1/port2)	Max time for bidirectional conv (s)	Max % packet loss (port1/port2)
PE3-DL2 failure						
	down	3/11	5/3	2/10	11	25%/50%
	up	0	0/4	0	0	0%/20%
P1-PE3 failure						
	down	5/5	4/3	5/3	5	25%/25%
	up	4/2	0/4	0/2	4	15%/10%
PE3 failure						
	down	2/6	3/5	3/5	6	15%/40%
	up	0	0	0	0	0%
P1 failure						
	down	20/21	20/22	19/21	22	50%/50%
	up	4/9	4/10	3/10	10	10%/25%

Core Protocol—OSPF

Table 7-7 PE-CE Protocol—OSPF Untuned

Scenario		T1 (s) (port1/port2)	T2 (s) (port1/port2)	T3 (s) (port1/port2)	Max time for bidirectional conv (s)	Max % packet loss (port1/port2)
PE3-DL2 failure						
	down	3/12	3/12	3/12	12	30%/50%
	up	0	0	0	0	0%
P1-PE3 failure						
	down	8/8	8/7	9/8	9	25%/25%
	up	0	0	0	0	0%

Table 7-7 PE-CE Protocol—OSPF Untuned

PE3 failure						
	down	0/6	0/5	0/4	6	0%/50%
	up	0	0	0	0	0%
P1 failure						
	down	5/9	8/8	7/10	10	25%/50%
	up	15/13	16/14	15/14	16	50%/50%

Table 7-8 PE-CE Protocol—OSPF Tuned

Scenario		T1 (s) (port1/port2)	T2 (s) (port1/port2)	T3 (s) (port1/port2)	Max time for bidirectional conv (s)	Max % packet loss (port1/port2)
PE3-DL2 failure						
	down	3/10	3/10	3/9	10	25%/50%
	up	0	0	0	0	0%
P1-PE3 failure						
	down	4/2	4/2	4/2	4	15%/5%
	up	0	0	0	0	0%
PE3 failure						
	down	0/9	0/9	0/8	9	0%/50%
	up	0	0	0	0	0%
P1 failure						
	down	0/5	0/5	0/5	5	0%/10%
	up	24/21	31/29	29/27	31	25%/25%

Table 7-9 PE-CE Protocol—EIGRP Untuned

Scenario		T1 (s) (port1/port2)	T2 (s) (port1/port2)	T3 (s) (port1/port2)	Max time for bidirectional conv (s)	Max % packet loss (port1/port2)
PE3-DL2 failure						
	down	3/11	3/11	3/12	12	35%/50%
	up	0	0	0	0	0%
P1-PE3 failure						
	down	9/8	9/7	9/7	9	25%/25%
	up	0	0	0	0	0%
PE3 failure						
	down	4/7	5/8	3/8	8	30%/40%
	up	0	0	0	0	0%

Table 7-9 PE-CE Protocol—EIGRP Untuned

P1 failure						
	down	9/11	8/10	8/9	11	50%/25%
	up	14/24	14/22	15/12	24	50%/50%

Table 7-10 PE-CE Protocol—EIGRP Tuned

Scenario		T1 (s) (port1/port2)	T2 (s) (port1/port2)	T3 (s) (port1/port2)	Max time for bidirectional conv (s)	Max % packet loss (port1/port2)
PE3-DL2 failure						
	down	3/11	3/11	3/10	11	25%/50%
	up	0	0	0	0	0%
P1-PE3 failure						
	down	4/2	5/3	5/2	5	25%/5%
	up	0	0	0	0	0%
PE3 failure						
	down	3/8	3/8	3/8	8	15%/50%
	up	0	0	0	0	0%
P1 failure						
	down	0	0/5	0	5	0%/20%
	up	27/26	25/22	27/26	27	50%/50%

CHAPTER 8

Configurations and Logs for Each VPN

This section includes the entire configurations from the Cisco 7600 and Cisco 12000 as PEs.

Cisco 7600 as PE

```
hostname 7600-DC2-PE3
!
boot system flash disk0:s72033-adventerprisek9_wan-mz.122-18.SXE2.bin
!
ip vrf blue-data
 rd 10:1053
 route-target export 10:105
 route-target import 10:105
 mdt default 239.232.10.1
!
ip vrf blue-voice
 rd 10:1063
 route-target export 10:106
 route-target import 10:106
 mdt default 239.232.10.2
!
ip vrf red-data
 rd 10:1033
 route-target export 10:103
 route-target import 10:103
 mdt default 239.232.10.3
 mdt data 239.232.20.32 0.0.0.15 threshold 1
!
ip vrf red-voice
 rd 10:1043
 route-target export 10:104
 route-target import 10:104
 mdt default 239.232.10.4
 mdt data 239.232.20.48 0.0.0.15 threshold 1
!
ip vrf sitemap
!
ip multicast-routing
ip multicast-routing vrf blue-data
ip multicast-routing vrf blue-voice
ip multicast-routing vrf red-data
ip multicast-routing vrf red-voice
no ip domain-lookup
vtp mode transparent
mpls label protocol ldp
```

```
tag-switching tdp router-id Loopback0 force
mls ip multicast replication-mode ingress
mls qos
!
interface Loopback0
 ip address 125.1.125.7 255.255.255.255
 ip pim sparse-mode
!
interface Loopback1
 ip vrf forwarding red-data
 ip address 125.1.125.103 255.255.255.255
!
interface GigabitEthernet1/1
 description To P1 - intf G4/0/0
 ip address 125.1.102.2 255.255.255.252
 ip pim sparse-mode
 wrr-queue bandwidth 5 25 70
 wrr-queue queue-limit 5 25 40
 wrr-queue random-detect min-threshold 1 80 100 100 100 100 100 100 100
 wrr-queue random-detect min-threshold 2 80 100 100 100 100 100 100 100
 wrr-queue random-detect min-threshold 3 50 60 70 80 90 100 100 100
 wrr-queue random-detect max-threshold 1 100 100 100 100 100 100 100 100
 wrr-queue random-detect max-threshold 2 100 100 100 100 100 100 100 100
 wrr-queue random-detect max-threshold 3 60 70 80 90 100 100 100 100
 wrr-queue cos-map 1 1 1
 wrr-queue cos-map 2 1 0
 wrr-queue cos-map 3 1 4
 wrr-queue cos-map 3 2 2
 wrr-queue cos-map 3 3 3
 wrr-queue cos-map 3 4 6
 wrr-queue cos-map 3 5 7
 tag-switching ip
 mls qos trust dscp
!
interface GigabitEthernet1/2
 description To P2 - intf G1/9
 ip address 125.1.102.14 255.255.255.252
 ip pim sparse-mode
 wrr-queue bandwidth 5 25 70
 wrr-queue queue-limit 5 25 40
 wrr-queue random-detect min-threshold 1 80 100 100 100 100 100 100 100
 wrr-queue random-detect min-threshold 2 80 100 100 100 100 100 100 100
 wrr-queue random-detect min-threshold 3 50 60 70 80 90 100 100 100
 wrr-queue random-detect max-threshold 1 100 100 100 100 100 100 100 100
 wrr-queue random-detect max-threshold 2 100 100 100 100 100 100 100 100
 wrr-queue random-detect max-threshold 3 60 70 80 90 100 100 100 100
 wrr-queue cos-map 1 1 1
 wrr-queue cos-map 2 1 0
 wrr-queue cos-map 3 1 4
 wrr-queue cos-map 3 2 2
 wrr-queue cos-map 3 3 3
 wrr-queue cos-map 3 4 6
 wrr-queue cos-map 3 5 7
 tag-switching ip
 mls qos trust dscp
!
interface GigabitEthernet1/3
 description To PE4 - intf TBD
 ip address 125.1.102.21 255.255.255.252
 ip pim sparse-mode
 wrr-queue bandwidth 5 25 70
 wrr-queue queue-limit 5 25 40
 wrr-queue random-detect min-threshold 1 80 100 100 100 100 100 100 100
 wrr-queue random-detect min-threshold 2 80 100 100 100 100 100 100 100
```

```
 wrr-queue random-detect min-threshold 3 50 60 70 80 90 100 100 100
 wrr-queue random-detect max-threshold 1 100 100 100 100 100 100 100 100
 wrr-queue random-detect max-threshold 2 100 100 100 100 100 100 100 100
 wrr-queue random-detect max-threshold 3 60 70 80 90 100 100 100 100
 wrr-queue cos-map 1 1 1
 wrr-queue cos-map 2 1 0
 wrr-queue cos-map 3 1 4
 wrr-queue cos-map 3 2 2
 wrr-queue cos-map 3 3 3
 wrr-queue cos-map 3 4 6
 wrr-queue cos-map 3 5 7
 tag-switching ip
 mls qos trust dscp
!
interface GigabitEthernet1/4
 description To DL2 - intf G5/1
 no ip address
 load-interval 30
 wrr-queue bandwidth 5 25 70
 wrr-queue queue-limit 5 25 40
 wrr-queue random-detect min-threshold 1 80 100 100 100 100 100 100 100
 wrr-queue random-detect min-threshold 2 80 100 100 100 100 100 100 100
 wrr-queue random-detect min-threshold 3 50 60 70 80 90 100 100 100
 wrr-queue random-detect max-threshold 1 100 100 100 100 100 100 100 100
 wrr-queue random-detect max-threshold 2 100 100 100 100 100 100 100 100
 wrr-queue random-detect max-threshold 3 60 70 80 90 100 100 100 100
 wrr-queue cos-map 1 1 1
 wrr-queue cos-map 2 1 0
 wrr-queue cos-map 3 1 4
 wrr-queue cos-map 3 2 2
 wrr-queue cos-map 3 3 3
 wrr-queue cos-map 3 4 6
 wrr-queue cos-map 3 5 7
 mls qos trust dscp
!
interface GigabitEthernet1/4.1
 description RED-DATA
 encapsulation dot1Q 161
 ip vrf forwarding red-data
 ip address 125.1.102.33 255.255.255.252
 ip pim sparse-mode
 wrr-queue bandwidth 5 25 70
 wrr-queue queue-limit 5 25 40
 wrr-queue random-detect min-threshold 1 80 100 100 100 100 100 100 100
 wrr-queue random-detect min-threshold 2 80 100 100 100 100 100 100 100
 wrr-queue random-detect min-threshold 3 50 60 70 80 90 100 100 100
 wrr-queue random-detect max-threshold 1 100 100 100 100 100 100 100 100
 wrr-queue random-detect max-threshold 2 100 100 100 100 100 100 100 100
 wrr-queue random-detect max-threshold 3 60 70 80 90 100 100 100 100
 wrr-queue cos-map 1 1 1
 wrr-queue cos-map 2 1 0
 wrr-queue cos-map 3 1 4
 wrr-queue cos-map 3 2 2
 wrr-queue cos-map 3 3 3
 wrr-queue cos-map 3 4 6
 wrr-queue cos-map 3 5 7
!
interface GigabitEthernet1/4.2
 description RED-VOICE
 encapsulation dot1Q 162
 ip vrf forwarding red-voice
 ip address 125.1.102.37 255.255.255.252
 ip pim sparse-mode
 wrr-queue bandwidth 5 25 70
```

```
 wrr-queue queue-limit 5 25 40
 wrr-queue random-detect min-threshold 1 80 100 100 100 100 100 100 100
 wrr-queue random-detect min-threshold 2 80 100 100 100 100 100 100 100
 wrr-queue random-detect min-threshold 3 50 60 70 80 90 100 100 100
 wrr-queue random-detect max-threshold 1 100 100 100 100 100 100 100 100
 wrr-queue random-detect max-threshold 2 100 100 100 100 100 100 100 100
 wrr-queue random-detect max-threshold 3 60 70 80 90 100 100 100 100
 wrr-queue cos-map 1 1 1
 wrr-queue cos-map 2 1 0
 wrr-queue cos-map 3 1 4
 wrr-queue cos-map 3 2 2
 wrr-queue cos-map 3 3 3
 wrr-queue cos-map 3 4 6
 wrr-queue cos-map 3 5 7
!
interface GigabitEthernet1/5
 description To DL3 - intf G5/1
 no ip address
 ip pim sparse-mode
 wrr-queue bandwidth 5 25 70
 wrr-queue queue-limit 5 25 40
 wrr-queue random-detect min-threshold 1 80 100 100 100 100 100 100 100
 wrr-queue random-detect min-threshold 2 80 100 100 100 100 100 100 100
 wrr-queue random-detect min-threshold 3 50 60 70 80 90 100 100 100
 wrr-queue random-detect max-threshold 1 100 100 100 100 100 100 100 100
 wrr-queue random-detect max-threshold 2 100 100 100 100 100 100 100 100
 wrr-queue random-detect max-threshold 3 60 70 80 90 100 100 100 100
 wrr-queue cos-map 1 1 1
 wrr-queue cos-map 2 1 0
 wrr-queue cos-map 3 1 4
 wrr-queue cos-map 3 2 2
 wrr-queue cos-map 3 3 3
 wrr-queue cos-map 3 4 6
 wrr-queue cos-map 3 5 7
 mls qos trust dscp
!
interface GigabitEthernet1/5.1
 description BLUE-DATA
 encapsulation dot1Q 163
 ip vrf forwarding blue-data
 ip vrf sitemap SoODL3
 ip address 125.1.102.41 255.255.255.252
 ip pim sparse-mode
 wrr-queue bandwidth 5 25 70
 wrr-queue queue-limit 5 25 40
 wrr-queue random-detect min-threshold 1 80 100 100 100 100 100 100 100
 wrr-queue random-detect min-threshold 2 80 100 100 100 100 100 100 100
 wrr-queue random-detect min-threshold 3 50 60 70 80 90 100 100 100
 wrr-queue random-detect max-threshold 1 100 100 100 100 100 100 100 100
 wrr-queue random-detect max-threshold 2 100 100 100 100 100 100 100 100
 wrr-queue random-detect max-threshold 3 60 70 80 90 100 100 100 100
 wrr-queue cos-map 1 1 1
 wrr-queue cos-map 2 1 0
 wrr-queue cos-map 3 1 4
 wrr-queue cos-map 3 2 2
 wrr-queue cos-map 3 3 3
 wrr-queue cos-map 3 4 6
 wrr-queue cos-map 3 5 7
!
interface GigabitEthernet1/5.2
 description BLUE-VOICE
 encapsulation dot1Q 164
 ip vrf forwarding blue-voice
 ip address 125.1.102.45 255.255.255.252
```

```
 ip pim sparse-mode
 wrr-queue bandwidth 5 25 70
 wrr-queue queue-limit 5 25 40
 wrr-queue random-detect min-threshold 1 80 100 100 100 100 100 100 100
 wrr-queue random-detect min-threshold 2 80 100 100 100 100 100 100 100
 wrr-queue random-detect min-threshold 3 50 60 70 80 90 100 100 100
 wrr-queue random-detect max-threshold 1 100 100 100 100 100 100 100 100
 wrr-queue random-detect max-threshold 2 100 100 100 100 100 100 100 100
 wrr-queue random-detect max-threshold 3 60 70 80 90 100 100 100 100
 wrr-queue cos-map 1 1 1
 wrr-queue cos-map 2 1 0
 wrr-queue cos-map 3 1 4
 wrr-queue cos-map 3 2 2
 wrr-queue cos-map 3 3 3
 wrr-queue cos-map 3 4 6
 wrr-queue cos-map 3 5 7
!
interface GigabitEthernet1/6
 description To SS2 - intf G3/1
 ip address 125.1.102.65 255.255.255.252
 wrr-queue cos-map 1 1 1
 wrr-queue cos-map 2 1 0
 wrr-queue cos-map 3 1 4
 wrr-queue cos-map 3 2 2
 wrr-queue cos-map 3 3 3
 wrr-queue cos-map 3 4 6
 wrr-queue cos-map 3 5 7
 tag-switching ip
 mls qos trust dscp
!
interface GigabitEthernet5/2
 ip address 172.26.185.109 255.255.255.0
 media-type rj45
 duplex full
!
router eigrp 1
 variance 2
 network 125.1.125.7 0.0.0.0
 network 125.0.0.0
 maximum-paths 8
 no auto-summary
!
router eigrp 10
 no auto-summary
 !
 address-family ipv4 vrf blue-voice
 variance 2
 redistribute bgp 1 metric 1000000 100 255 1 1500
 network 125.1.2.128 0.0.0.127
 network 125.1.102.44 0.0.0.3
 maximum-paths 8
 no auto-summary
 autonomous-system 11
 exit-address-family
 !
 address-family ipv4 vrf blue-data
 variance 2
 redistribute bgp 1 metric 1000000 100 255 1 1500
 network 125.1.2.0 0.0.0.127
 network 125.1.102.40 0.0.0.3
 maximum-paths 8
 distance eigrp 210 210
 no auto-summary
 autonomous-system 10
```

```
 exit-address-family
!
router ospf 1 vrf red-data
 log-adjacency-changes
 area 0 sham-link 125.1.125.103 125.1.125.108
 area 0 sham-link 125.1.125.103 125.1.125.107
 redistribute bgp 1 subnets
 network 125.1.102.32 0.0.0.3 area 0
!
router ospf 2 vrf red-voice
 log-adjacency-changes
 redistribute bgp 1 subnets
 network 125.1.102.36 0.0.0.3 area 0
!
router bgp 1
 no synchronization
 bgp log-neighbor-changes
 network 125.1.125.103 mask 0.0.0.0
 neighbor 125.1.125.15 remote-as 1
 neighbor 125.1.125.15 update-source Loopback0
 neighbor 125.1.125.16 remote-as 1
 neighbor 125.1.125.16 update-source Loopback0
 no auto-summary
 !
 address-family vpnv4
 neighbor 125.1.125.15 activate
 neighbor 125.1.125.15 send-community extended
 neighbor 125.1.125.16 activate
 neighbor 125.1.125.16 send-community extended
 exit-address-family
 !
 address-family ipv4 vrf sitemap
 no auto-summary
 no synchronization
 exit-address-family
 !
 address-family ipv4 vrf red-voice
 redistribute ospf 2 vrf red-voice match internal external 1 external 2
 no auto-summary
 no synchronization
 exit-address-family
 !
 address-family ipv4 vrf red-data
 redistribute connected metric 1
 redistribute ospf 1 vrf red-data match internal external 1 external 2
 maximum-paths ibgp  6
 no auto-summary
 no synchronization
 exit-address-family
 !
 address-family ipv4 vrf blue-voice
 redistribute eigrp 11
 maximum-paths ibgp unequal-cost 8
 no auto-summary
 no synchronization
 exit-address-family
 !
 address-family ipv4 vrf blue-data
 redistribute eigrp 10
 maximum-paths ibgp unequal-cost 8
 no auto-summary
 no synchronization
 exit-address-family
!
```

```
ip classless
ip route 0.0.0.0 0.0.0.0 172.26.185.1
!
no ip http server
ip pim ssm range 1
ip pim vrf blue-data rp-address 1.1.1.11 override
ip pim vrf blue-voice rp-address 2.2.2.11 override
ip pim vrf red-data rp-address 3.3.3.11 override
ip pim vrf red-voice rp-address 4.4.4.11 override
!
ip access-list standard mdtacl
 permit 239.232.10.0 0.0.0.255
 permit 239.232.20.0 0.0.0.255
ip access-list standard orga
 permit 224.2.131.239
 permit 224.2.149.231
 permit 224.2.159.191
 permit 224.2.201.231
 permit 239.193.10.252
!
access-list 1 permit 239.232.0.0 0.0.255.255
!
route-map SoODL3 permit 10
 set extcommunity soo 3:1
!
```

Cisco 12000 as PE

```
hostname 12k-DC2-PE4
!
boot-start-marker
boot system flash disk0:c12kprp-k4p-mz.120-31.S.bin
boot-end-marker
!
hw-module slot 2 qos interface queues 8
!
ip vrf blue-data
 rd 10:1054
 route-target export 10:105
 route-target import 10:105
 mdt default 239.232.10.1
 mdt data 239.232.20.0 0.0.0.15 threshold 1
!
ip vrf blue-voice
 rd 10:1064
 route-target export 10:106
 route-target import 10:106
 mdt default 239.232.10.2
 mdt data 239.232.20.16 0.0.0.15 threshold 1
!
ip vrf red-data
 rd 10:1034
 route-target export 10:103
 route-target import 10:103
 mdt default 239.232.10.3
 mdt data 239.232.20.32 0.0.0.15 threshold 1
!
ip vrf red-voice
 rd 10:1044
 route-target export 10:104
 route-target import 10:104
```

```
 mdt default 239.232.10.4
 mdt data 239.232.20.48 0.0.0.15 threshold 1
!
ip multicast-routing distributed
ip multicast-routing vrf blue-data distributed
ip multicast-routing vrf blue-voice distributed
ip multicast-routing vrf red-data distributed
ip multicast-routing vrf red-voice distributed
!
class-map match-any realtime
  match mpls experimental  5
class-map match-any realtime-2ce
  match qos-group 5
class-map match-all red-voice
  match vlan  166
class-map match-any bulk-data
  match mpls experimental  1
class-map match-any network-control-2ce
  match qos-group 7
class-map match-any interwork-control
  match mpls experimental  6
class-map match-any network-control
  match mpls experimental  7
class-map match-any bulk-data-2ce
  match qos-group 1
class-map match-any interwork-control-2ce
  match qos-group 6
  match  precedence 6
class-map match-any bus-critical
  match mpls experimental  3
class-map match-any trans-data
  match mpls experimental  2
class-map match-all red-data
  match vlan  165
class-map match-any bus-critical-2ce
  match qos-group 3
class-map match-any trans-data-2ce
  match qos-group 2
class-map match-any video-2ce
  match qos-group 4
class-map match-any video
  match mpls experimental  4
!
policy-map q-core-out
  class realtime
    priority
   police cir percent 30 bc 500 ms conform-action transmit  exceed-action drop
  class network-control
    bandwidth remaining percent 7
    random-detect
    random-detect precedence 7 625 packets 4721 packets 1
  class interwork-control
    bandwidth remaining percent 7
    random-detect
    random-detect precedence 6 625 packets 4721 packets 1
  class bus-critical
    bandwidth remaining percent 14
    random-detect
    random-detect precedence 3 625 packets 4721 packets 1
  class trans-data
    bandwidth remaining percent 14
    random-detect
    random-detect precedence 2 625 packets 4721 packets 1
  class video
```

```
    bandwidth remaining percent 14
    random-detect
    random-detect precedence 4 625 packets 4721 packets 1
  class bulk-data
    bandwidth remaining percent 7
    random-detect
    random-detect precedence 1 625 packets 4721 packets 1
  class class-default
    bandwidth remaining percent 36
    random-detect
    random-detect precedence 0 625 packets 4721 packets 1
policy-map q-2ce-out-1
  class network-control-2ce
    bandwidth percent 7
    random-detect discard-class-based
    random-detect discard-class 7 625 packets 4721 packets 1
  class interwork-control-2ce
    bandwidth percent 7
    random-detect discard-class-based
    random-detect discard-class 6 625 packets 4721 packets 1
  class bus-critical-2ce
    bandwidth percent 14
    random-detect discard-class-based
    random-detect discard-class 3 625 packets 4721 packets 1
  class trans-data-2ce
    bandwidth percent 14
    random-detect discard-class-based
    random-detect discard-class 2 625 packets 4721 packets 1
  class video-2ce
    bandwidth percent 14
    random-detect discard-class-based
    random-detect discard-class 4 625 packets 4721 packets 1
  class bulk-data-2ce
    bandwidth percent 7
    random-detect discard-class-based
    random-detect discard-class 1 625 packets 4721 packets 1
  class class-default
    bandwidth percent 36
    random-detect discard-class-based
    random-detect discard-class 0 625 packets 4721 packets 1
policy-map q-2ce-out-2
  class realtime-2ce
    priority
   police cir percent 95 bc 500 ms conform-action transmit  exceed-action drop
  class interwork-control-2ce
    bandwidth percent 3
    random-detect
    random-detect precedence 6 4720 packets 4721 packets 1
  class class-default
    bandwidth percent 2
    random-detect
policy-map q-2ce-out-parent
  class red-data
    shape average percent 60
   service-policy q-2ce-out-1
  class red-voice
    shape average percent 40
   service-policy q-2ce-out-2
policy-map egr-pe-in
  class realtime
   set qos-group 5
   set discard-class 5
  class network-control
   set qos-group 7
```

```
   set discard-class 7
  class interwork-control
   set qos-group 6
   set discard-class 6
  class bus-critical
   set qos-group 3
   set discard-class 3
  class trans-data
   set qos-group 2
   set discard-class 2
  class video
   set qos-group 4
   set discard-class 4
  class bulk-data
   set qos-group 1
   set discard-class 1
!
mpls label protocol ldp
tag-switching tdp router-id Loopback0 force
!
interface Loopback0
 ip address 125.1.125.8 255.255.255.255
 no ip directed-broadcast
 ip pim sparse-mode
!
interface Loopback1
 ip vrf forwarding red-data
 ip address 125.1.125.104 255.255.255.255
 no ip directed-broadcast
!
interface GigabitEthernet2/0
 description To DL2 - intf G5/2
 no ip address
 no ip directed-broadcast
 load-interval 30
 negotiation auto
 service-policy output q-2ce-out-parent
!
interface GigabitEthernet2/0.1
 description RED-DATA
 encapsulation dot1Q 165
 ip vrf forwarding red-data
 ip address 125.1.102.49 255.255.255.252
 no ip directed-broadcast
 ip pim sparse-mode
!
interface GigabitEthernet2/0.2
 description RED-VOICE
 encapsulation dot1Q 166
 ip vrf forwarding red-voice
 ip address 125.1.102.53 255.255.255.252
 no ip directed-broadcast
 ip pim sparse-mode
!
interface GigabitEthernet2/1
 description To DL3 - intf G5/2
 no ip address
 no ip directed-broadcast
 negotiation auto
!
interface GigabitEthernet2/1.1
 description BLUE-DATA
 encapsulation dot1Q 167
 ip vrf forwarding blue-data
```

```
 ip vrf sitemap SoODL3
 ip address 125.1.102.57 255.255.255.252
 no ip directed-broadcast
 ip pim sparse-mode
 service-policy output q-2ce-out-data
!
interface GigabitEthernet2/1.2
 description BLUE-VOICE
 encapsulation dot1Q 168
 ip vrf forwarding blue-voice
 ip address 125.1.102.61 255.255.255.252
 no ip directed-broadcast
 ip pim sparse-mode
!
interface GigabitEthernet3/0/0
 description To SS2 - intf G3/2
 ip address 125.1.102.69 255.255.255.252
 no ip directed-broadcast
 negotiation auto
 tag-switching ip
 service-policy input egr-pe-in
 service-policy output q-core-out
!
interface GigabitEthernet3/0/1
 description To P2 - intf G1/10
 ip address 125.1.102.18 255.255.255.252
 no ip directed-broadcast
 ip pim sparse-mode
 load-interval 30
 negotiation auto
 tag-switching ip
 service-policy input egr-pe-in
 service-policy output q-core-out
!
interface GigabitEthernet3/0/2
 description To PE3 - intf G1/3
 ip address 125.1.102.22 255.255.255.252
 no ip directed-broadcast
 ip pim sparse-mode
 negotiation auto
 tag-switching ip
 service-policy input egr-pe-in
 service-policy output q-core-out
!
interface GigabitEthernet3/3/0
 description To P1 - intf G4/0/2
 ip address 125.1.102.6 255.255.255.252
 no ip directed-broadcast
 ip pim sparse-mode
 load-interval 30
 negotiation auto
 tag-switching ip
 service-policy input egr-pe-in
 service-policy output q-core-out
!
interface Ethernet2
 description Mgmt
 ip address 172.26.185.108 255.255.255.0
 no ip directed-broadcast
 no ip route-cache
 no ip mroute-cache
!
router eigrp 1
 !
```

```
 address-family ipv4
 variance 2
 network 125.1.125.8 0.0.0.0
 network 125.0.0.0
 maximum-paths 8
 exit-address-family
!
router eigrp 10
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv4 vrf blue-voice
 variance 2
 redistribute bgp 1 metric 1000000 100 255 1 1500
 network 125.1.2.128 0.0.0.127
 network 125.1.102.60 0.0.0.3
 maximum-paths 8
 autonomous-system 11
 exit-address-family
 !
 address-family ipv4 vrf blue-data
 variance 2
 redistribute bgp 1 metric 1000000 100 255 1 1500
 network 125.1.2.0 0.0.0.127
 network 125.1.102.56 0.0.0.3
 maximum-paths 8
 distance eigrp 210 210
 autonomous-system 10
 exit-address-family
!
router ospf 1 vrf red-data
 log-adjacency-changes
 area 0 sham-link 125.1.125.104 125.1.125.107
 area 0 sham-link 125.1.125.104 125.1.125.108
 redistribute bgp 1 subnets
 network 125.1.102.48 0.0.0.3 area 0
!
router ospf 2 vrf red-voice
 log-adjacency-changes
 redistribute bgp 1 subnets
 network 125.1.102.52 0.0.0.3 area 0
!
router bgp 1
 no synchronization
 bgp log-neighbor-changes
 neighbor 125.1.125.15 remote-as 1
 neighbor 125.1.125.15 update-source Loopback0
 neighbor 125.1.125.16 remote-as 1
 neighbor 125.1.125.16 update-source Loopback0
 no auto-summary
 !
 address-family ipv4 mdt
 neighbor 125.1.125.15 activate
 neighbor 125.1.125.15 send-community extended
 neighbor 125.1.125.16 activate
 neighbor 125.1.125.16 send-community extended
 exit-address-family
 !
 address-family vpnv4
 neighbor 125.1.125.15 activate
 neighbor 125.1.125.15 send-community extended
 neighbor 125.1.125.16 activate
 neighbor 125.1.125.16 send-community extended
```

```
 exit-address-family
 !
 address-family ipv4 vrf red-voice
 redistribute ospf 2 vrf red-voice match internal external 1 external 2
 maximum-paths ibgp unequal-cost 6
 no auto-summary
 no synchronization
 exit-address-family
 !
 address-family ipv4 vrf red-data
 redistribute connected metric 1
 redistribute ospf 1 vrf red-data match internal external 1 external 2
 maximum-paths ibgp 2
 no auto-summary
 no synchronization
 exit-address-family
 !
 address-family ipv4 vrf blue-voice
 redistribute eigrp 11
 maximum-paths ibgp unequal-cost 8
 no auto-summary
 no synchronization
 exit-address-family
 !
 address-family ipv4 vrf blue-data
 redistribute eigrp 10
 maximum-paths ibgp unequal-cost 8
 no auto-summary
 no synchronization
 exit-address-family
 !
ip classless
ip route 0.0.0.0 0.0.0.0 172.26.185.1
!
ip pim ssm range 1
ip pim vrf blue-data rp-address 1.1.1.11 override
ip pim vrf blue-voice rp-address 2.2.2.11 override
ip pim vrf red-data rp-address 3.3.3.11 override
ip pim vrf red-voice rp-address 4.4.4.11 override
!
ip access-list standard mdtacl
 permit 239.232.10.0 0.0.0.255
 permit 239.232.20.0 0.0.0.255
ip access-list standard orga
 permit 224.2.131.239
 permit 224.2.149.231
 permit 224.2.159.191
 permit 224.2.201.231
 permit 239.193.10.252
rx-cos-slot all SLOT_TABLE
!
slot-table-cos SLOT_TABLE
 destination-slot all GE_policy
 multicast GE-multicast
!
cos-queue-group GE-multicast
 precedence 4 queue 4
!
cos-queue-group GE_policy
 precedence 0 random-detect-label 1
 precedence 1 queue 1
 precedence 1 random-detect-label 1
 precedence 2 queue 2
 precedence 2 random-detect-label 1
```

```
 precedence 3 queue 3
 precedence 3 random-detect-label 1
 precedence 4 queue 4
 precedence 4 random-detect-label 1
 precedence 5 queue low-latency
 precedence 6 queue 5
 precedence 6 random-detect-label 1
 precedence 7 queue 6
 precedence 7 random-detect-label 1
 random-detect-label 1 625 4721 1
 queue 1 2
 queue 2 4
 queue 3 4
 queue 4 4
 queue 5 2
 queue 6 2
 queue low-latency strict-priority
access-list 1 permit 239.232.0.0 0.0.255.255
route-map SoODL3 permit 10
 set extcommunity soo 3:1
```

Service Validation

Core Verification

1. Ping tests to show each core device is accessible.

Threre are four core routers, P1….P4. 10 PEs, and two Route Reflectors.

All of these devices should be able to communicate. A good test would be to run ping tests from one of the Route Reflectors. (Notice 125.1.125.1 to .4 are Ps, .5 to .14 are PEs and .15 is RR1 and .16 is RR2).

```
7200-DC2-RR1#ping 125.1.125.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 125.1.125.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
7200-DC2-RR1#ping 125.1.125.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 125.1.125.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
7200-DC2-RR1#ping 125.1.125.3

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 125.1.125.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
7200-DC2-RR1#ping 125.1.125.4

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 125.1.125.4, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
7200-DC2-RR1#ping 125.1.125.5

Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 125.1.125.5, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
7200-DC2-RR1#ping 125.1.125.6

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 125.1.125.6, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
7200-DC2-RR1#ping 125.1.125.7

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 125.1.125.7, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
7200-DC2-RR1#ping 125.1.125.8

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 125.1.125.8, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
7200-DC2-RR1#ping 125.1.125.9

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 125.1.125.9, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
7200-DC2-RR1#ping 125.1.125.10

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 125.1.125.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
7200-DC2-RR1#ping 125.1.125.11

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 125.1.125.11, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
7200-DC2-RR1#ping 125.1.125.12

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 125.1.125.12, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
7200-DC2-RR1#ping 125.1.125.13

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 125.1.125.13, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
7200-DC2-RR1#ping 125.1.125.14

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 125.1.125.14, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
7200-DC2-RR1#ping 125.1.125.15

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 125.1.125.15, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
7200-DC2-RR1#ping 125.1.125.16
```

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 125.1.125.16, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
7200-DC2-RR1#
```

2. Examine global table to verify no VPN routes exists in the global table:

Notice 125.1.101.x is used for VPN links in DC1. 125.1.102.x is used for VPN links in DC2, 125.1.103.x is used for VPN links in LC, 125.1.104.x for VPN links in MC, and 125.1.105.x for VPN links in SC1 which should not exist in the global table.

```
7600-LC-P4#sh ip ro 125.1.101.0
% Subnet not in table
7600-LC-P4#

7600-LC-P4#sh ip ro 125.1.104.0
% Subnet not in table
7600-LC-P4#

7600-LC-P4#sh ip ro 125.1.105.0
% Subnet not in table
7600-LC-P4#
```

Notice that no VRF information is stored on the core routers:

```
7600-LC-P4#sh ip vrf

7600-LC-P4#
```

3. Examine LFIBs to ensure that they have labels for all the prefixes.

For example, PE1 to P3 Interface:

```
7600-LC-P4#   sh mpls forwarding-table 125.1.100.32
Local  Outgoing    Prefix              Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id        switched   interface
36     33          125.1.100.32/30   0          Gi1/6      125.1.100.89
       33          125.1.100.32/30   0          Gi1/7      125.1.100.93
       50          125.1.100.32/30   0          Gi1/2      125.1.100.29
       50          125.1.100.32/30   0          Gi1/1      125.1.100.25
```

PE7 to P1 Interface:

```
7600-LC-P4#   sh mpls forwarding-table 125.1.100.0
Local  Outgoing    Prefix              Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id        switched   interface
43     Pop tag     125.1.100.0/30    0          Gi1/6      125.1.100.89
       Pop tag     125.1.100.0/30    0          Gi1/7      125.1.100.93
```

Full LFIB table can be viewed using the sh mpls forwarding-table command:

```
7600-LC-P4#sh mpls forwarding-table
Local  Outgoing    Prefix              Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id        switched   interface
16     Pop tag     125.1.125.10/32   26684      Gi1/8      125.1.103.22
17     Pop tag     125.1.103.4/30    0          Gi1/7      125.1.100.93
       Pop tag     125.1.103.4/30    0          Gi1/6      125.1.100.89
       Pop tag     125.1.103.4/30    0          Gi1/8      125.1.103.22
18     Pop tag     125.1.125.3/32    2437       Gi1/6      125.1.100.89
```

```
            Pop tag     125.1.125.3/32    0          Gi1/7      125.1.100.93
```

…….etc.

4. Ping tests to VPN sites and interfaces to verify that the core does not have access to the VPN sites.

Edge Verification

1. Verify VPNv4 peering with PEs and RRs.
2. Verify the VPN site routing table on the distribution layer device; check that these routes get propagated to VRF tables and BGP tables on ingress and egress PEs. Verify that the labels assigned by the ingress PEs are propagated to and used by the egress PEs.
3. Verify that multiple paths are installed in the forwarding table without any routing loops.
4. Verify that you can reach pertinent VPN sites from PEs and that multiple paths are used.

Baseline MPLS VPN

The VRF red-data on PE3 is used as the reference for highlighting the service validation steps, which are useful in troubleshooting as well. The following example examines the reachability to 3.3.3.13/32, which resides in DC1.

1. Verify the route exists in the VRF red-data routing table:

```
7600-DC2-PE3#sh ip route vrf red-data 3.3.3.13
Routing entry for 3.3.3.13/32
  Known via "bgp 1", distance 200, metric 2, type internal
  Redistributing via ospf 1
  Advertised by ospf 1 subnets
  Last update from 125.1.125.6 00:13:13 ago
  Routing Descriptor Blocks:
  * 125.1.125.5 (Default-IP-Routing-Table), from 125.1.125.15, 00:13:13 ago
      Route metric is 2, traffic share count is 1
      AS Hops 0
    125.1.125.6 (Default-IP-Routing-Table), from 125.1.125.15, 00:13:13 ago
      Route metric is 2, traffic share count is 1
      AS Hops 0
```

2. Verify that the nex-hop exists in the global routing table:

```
7600-DC2-PE3#sh ip route 125.1.125.6
Routing entry for 125.1.125.6/32
  Known via "eigrp 1", distance 90, metric 131072, type internal
  Redistributing via eigrp 1
  Last update from 125.1.102.13 on GigabitEthernet1/2, 23:59:59 ago
  Routing Descriptor Blocks:
  * 125.1.102.13, from 125.1.102.13, 23:59:59 ago, via GigabitEthernet1/2
      Route metric is 131072, traffic share count is 1
      Total delay is 5020 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 2

7600-DC2-PE3#sh ip route 125.1.125.5
Routing entry for 125.1.125.5/32
  Known via "eigrp 1", distance 90, metric 131072, type internal
  Redistributing via eigrp 1
  Last update from 125.1.102.1 on GigabitEthernet1/1, 1d00h ago
  Routing Descriptor Blocks:
```

```
  * 125.1.102.1, from 125.1.102.1, 1d00h ago, via GigabitEthernet1/1
      Route metric is 131072, traffic share count is 1
      Total delay is 5020 microseconds, minimum bandwidth is 1000000 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 2
```

3. Additional verification may include checking the BGP table for the VPN route:

```
7600-DC2-PE3#sh ip bgp vpnv4 vrf red-data 3.3.3.13
BGP routing table entry for 10:1033:3.3.3.13/32, version 266
Paths: (2 available, best #1, table red-data)
Multipath: iBGP
  Not advertised to any peer
  Local, imported path from 10:1031:3.3.3.13/32
    125.1.125.5 (metric 131072) from 125.1.125.15 (125.1.125.15)
      Origin incomplete, metric 2, localpref 100, valid, internal, multipath, best
      Extended Community: RT:10:103 OSPF DOMAIN ID:0x0005:0x000000010200
        OSPF RT:0.0.0.0:2:0 OSPF ROUTER ID:125.1.101.1:512
      Originator: 125.1.125.5, Cluster list: 125.1.125.15
  Local, imported path from 10:1032:3.3.3.13/32
    125.1.125.6 (metric 131072) from 125.1.125.15 (125.1.125.15)
      Origin incomplete, metric 2, localpref 100, valid, internal, multipath
      Extended Community: RT:10:103 OSPF DOMAIN ID:0x0005:0x000000010200
        OSPF RT:0.0.0.0:2:0 OSPF ROUTER ID:125.1.101.9:512
      Originator: 125.1.125.6, Cluster list: 125.1.125.15
```

4. Verify that the CEF entry exists for the VRF route (including the dual label stack):

```
7600-DC2-PE3#sh ip cef vrf red-data 3.3.3.13
3.3.3.13/32, version 34, epoch 0
0 packets, 0 bytes
  tag information set, all rewrites owned
    local tag: VPN-route-head
  via 125.1.125.5, 0 dependencies, recursive
    traffic share 1
    next hop 125.1.102.1, GigabitEthernet1/1 via 125.1.125.5/32 (Default)
    valid adjacency
    tag rewrite with Gi1/1, 125.1.102.1, tags imposed: {62 74}
  via 125.1.125.6, 0 dependencies, recursive
    traffic share 1
    next hop 125.1.102.13, GigabitEthernet1/2 via 125.1.125.6/32 (Default)
    valid adjacency
    tag rewrite with Gi1/2, 125.1.102.13, tags imposed: {16 73}
  0 packets, 0 bytes switched through the prefix
  tmstats: external 0 packets, 0 bytes
           internal 0 packets, 0 bytes
```

5. The top label information can also be verified by looking at the MPLS forwarding table:

```
7600-DC2-PE3#sh mpls for 125.1.125.5
Local  Outgoing    Prefix              Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id        switched   interface
89     62          125.1.125.5/32    0          Gi1/1      125.1.102.1
7600-DC2-PE3#sh mpls for 125.1.125.6
Local  Outgoing    Prefix              Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id        switched   interface
47     16          125.1.125.6/32    0          Gi1/2      125.1.102.13
```

6. The PE-to-PE LSP check can be done by using an LSP ping. A broken LSP can be detected by identifying the break point in the pings (this avoids hop-by-hop troubleshooting):

```
7600-DC2-PE3#ping mpls ip 125.1.125.5/32 ver
Sending 5, 100-byte MPLS Echos to 125.1.125.5/32,
     timeout is 2 seconds, send interval is 0 msec:
```

```
Codes: '!' - success, 'Q' - request not transmitted,
       '.' - timeout, 'U' - unreachable,
       'R' - downstream router but not target,
       'M' - malformed request

Type escape sequence to abort.
!     125.1.100.62, return code 3
!     125.1.100.62, return code 3
!     125.1.100.62, return code 3
!     125.1.100.62, return code 3
!     125.1.100.62, return code 3

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
```

7. If there is a valid source address present on the PE, a VRF ping can be performed to check the destination address reachability from the PE.

```
7600-DC2-PE3# ping vrf red-data ip 3.3.3.13

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 3.3.3.13, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
```

8. If packet drops are being experienced and the source of the drop has been identified, the above steps can be performed. Additionally, the following **show** commands may be useful:

```
7600-DC2-PE3#sh cef drop
CEF Drop Statistics
Slot  Encap_fail  Unresolved Unsupported    No_route      No_adj  ChkSum_Err
RP             5           0           0           0           0           0
5              0           0           0           0           0           0
1              0           0           0           0           0           0
6              0           0           0           0           0           0
25             0           0           0           0           0           0
IPv6 CEF Drop Statistics
Slot  Encap_fail  Unresolved Unsupported    No_route      No_adj
RP             0           0           0           0           0

7600-DC2-PE3#sh cef not-cef-switched
CEF Packets passed on to next switching layer
Slot  No_adj No_encap Unsupp'ted Redirect  Receive  Options   Access     Frag
RP         0       0           1        0    54016        0        0        0
5          0       0           0        0        0        0        0        0
1          0       0           0        0        0        0        0        0
6          0       0           0        0        0        0        0        0
25         0       0           0        0        0        0        0        0
IPv6 CEF Packets passed on to next switching layer
Slot  No_adj No_encap Unsupp'ted Redirect  Receive  Options   Access      MTU
RP         0       0           0        0        0        0        0        0
```

For checking whether the locally originated route is advertised to remote PEs correctly, the following steps may be performed:

1. Toward the CE side, verify that the PE-CE routing protocol is up and the neighbor relationship formed:

```
7600-DC2-PE3#sh ip ospf nei g1/4.1

Neighbor ID     Pri   State           Dead Time   Address         Interface
3.3.3.12          1   FULL/DR         00:00:32    125.1.102.34    GigabitEthernet1/4.1
```

2. Verify that the route is in the local VRF table:

```
7600-DC2-PE3#sh ip route vrf red-data 3.3.3.12
Routing entry for 3.3.3.12/32
  Known via "ospf 1", distance 110, metric 2, type intra area
  Redistributing via bgp 1
  Advertised by bgp 1 match internal external 1 & 2
  Last update from 125.1.102.34 on GigabitEthernet1/4.1, 00:27:03 ago
  Routing Descriptor Blocks:
  * 125.1.102.34, from 3.3.3.12, 00:27:03 ago, via GigabitEthernet1/4.1
      Route metric is 2, traffic share count is 1
```

3. Because you are redistributing the OSPF-learned routes into BGP, ensure that the route exists in the BGP table and is marked for advertisement:

```
7600-DC2-PE3#sh ip bgp vpnv4 vrf red-data 3.3.3.12
BGP routing table entry for 10:1033:3.3.3.12/32, version 16756
Paths: (2 available, best #2, table red-data)
Multipath: iBGP
  Advertised to update-groups:
     1
  Local, imported path from 10:1034:3.3.3.12/32
    125.1.125.8 (metric 130816) from 125.1.125.15 (125.1.125.15)
      Origin incomplete, metric 2, localpref 100, valid, internal
      Extended Community: RT:10:103 OSPF DOMAIN ID:0x0005:0x000000010200
        OSPF RT:0.0.0.0:2:0 OSPF ROUTER ID:125.1.125.104:0
      Originator: 125.1.125.8, Cluster list: 125.1.125.15
  Local
    125.1.102.34 from 0.0.0.0 (125.1.125.7)
      Origin incomplete, metric 2, localpref 100, weight 32768, valid, sourced, best
      Extended Community: RT:10:103 OSPF DOMAIN ID:0x0005:0x000000010200
        OSPF RT:0.0.0.0:2:0 OSPF ROUTER ID:125.1.125.103:512
```

OSPF Backdoor Link Verifications

Traceroute results from DL2 to DL6 demonstrate that without the backdoor link, traffic flows over an MPLS network:

```
DL2-DC2-RED#traceroute vrf red-data 1.1.1.16

Type escape sequence to abort.
Tracing the route to 1.1.1.16

  1 125.1.102.49 0 msec
    125.1.102.33 0 msec
    125.1.102.49 0 msec
  2 125.1.102.1 [MPLS: Labels 52/76 Exp 0] 4 msec
    125.1.102.17 [MPLS: Labels 63/76 Exp 0] 0 msec
    125.1.102.1 [MPLS: Labels 52/76 Exp 0] 0 msec
  3 125.1.104.9 [MPLS: Label 76 Exp 0] 0 msec
    125.1.104.1 [MPLS: Label 76 Exp 0] 0 msec
    125.1.104.9 [MPLS: Label 76 Exp 0] 0 msec
  4 125.1.104.2 4 msec
    125.1.104.10 0 msec *
DL2-DC2-RED#
```

Traceroute from DL2 to DL1 demonstrates that without the backdoor link, traffic flows over an MPLS network:

```
DL2-DC2-RED# traceroute vrf red-data 1.1.1.1

Type escape sequence to abort.
Tracing the route to 1.1.1.1
```

```
  1 125.1.102.49 0 msec
    125.1.102.33 0 msec
    125.1.102.49 4 msec
  2 125.1.102.13 [MPLS: Labels 54/20 Exp 0] 0 msec
    125.1.102.17 [MPLS: Labels 54/20 Exp 0] 0 msec
    125.1.102.13 [MPLS: Labels 54/20 Exp 0] 0 msec
  3 125.1.101.9 [MPLS: Label 20 Exp 0] 0 msec 0 msec 0 msec
  4 125.1.101.10 4 msec *  0 msec

       DL2-DC2-RED#sh ip ro vrf red-data

Routing Table: red-data

Gateway of last resort is not set

     1.0.0.0/32 is subnetted, 3 subnets
O IA    1.1.1.1 [110/3] via 125.1.102.49, 01:29:09, GigabitEthernet5/2.1
                [110/3] via 125.1.102.33, 01:29:09, GigabitEthernet5/1.1
O IA    1.1.1.16 [110/3] via 125.1.102.49, 01:29:09, GigabitEthernet5/2.1
                 [110/3] via 125.1.102.33, 01:29:09, GigabitEthernet5/1.1
```

Traceroute from DL1 to DL6 demonstrates that without the backdoor link, traffic flows over an MPLS network:

```
DL1-DC1-RED#traceroute vrf red-data 1.1.1.16

Type escape sequence to abort.
Tracing the route to 1.1.1.16

  1 125.1.101.1 0 msec
    125.1.101.9 4 msec
    125.1.101.1 0 msec
  2 125.1.100.73 [MPLS: Labels 32/76 Exp 0] 0 msec
    125.1.100.57 [MPLS: Labels 52/76 Exp 0] 0 msec
    125.1.100.73 [MPLS: Labels 32/76 Exp 0] 0 msec
  3 125.1.104.1 [MPLS: Label 76 Exp 0] 0 msec
    125.1.100.38 [MPLS: Labels 24/76 Exp 0] 4 msec
    125.1.104.1 [MPLS: Label 76 Exp 0] 0 msec
  4 125.1.104.1 [MPLS: Label 76 Exp 0] 0 msec
    125.1.104.2 0 msec
    125.1.104.1 [MPLS: Label 76 Exp 0] 0 msec

DL6-MC-RED#  sh ip ro (all the routes except for C and all the remote VPN sites are IA,
125.1.141.0/24- .148 are E):

Gateway of last resort is 172.26.185.1 to network 0.0.0.0

     1.0.0.0/32 is subnetted, 3 subnets
O IA    1.1.1.1 [110/3] via 125.1.104.9, 1d04h, GigabitEthernet2/2
                [110/3] via 125.1.104.1, 1d04h, GigabitEthernet2/1
C       1.1.1.16 is directly connected, Loopback0
O IA    1.1.1.23 [110/3] via 125.1.104.1, 2d01h, GigabitEthernet2/1
                 [110/3] via 125.1.104.9, 2d01h, GigabitEthernet2/2
     100.0.0.0/30 is subnetted, 2560 subnets
O IA    100.1.37.64 [110/5] via 125.1.104.9, 02:59:39, GigabitEthernet2/2
                    [110/5] via 125.1.104.1, 02:59:39, GigabitEthernet2/1


                        [110/20] via 125.1.104.1, 01:49:03, GigabitEthernet2/1
O E2    125.1.147.0/24 [110/20] via 125.1.104.9, 01:49:03, GigabitEthernet2/2
                       [110/20] via 125.1.104.1, 01:49:03, GigabitEthernet2/1
O E2    125.1.141.0/24 [110/20] via 125.1.104.9, 01:49:03, GigabitEthernet2/2
                       [110/20] via 125.1.104.1, 01:49:03, GigabitEthernet2/1
O E2    125.1.142.0/24 [110/20] via 125.1.104.9, 01:49:03, GigabitEthernet2/2
```

```
                     [110/20] via 125.1.104.1, 01:49:03, GigabitEthernet2/1
     12.0.0.0/27 is subnetted, 2 subnets
O IA    12.0.13.0 [110/4] via 125.1.104.9, 03:01:50, GigabitEthernet2/2
                  [110/4] via 125.1.104.1, 03:01:50, GigabitEthernet2/1
O IA    12.0.9.0 [110/4] via 125.1.104.1, 03:01:56, GigabitEthernet2/1
                 [110/4] via 125.1.104.9, 03:01:56, GigabitEthernet2/2
```

With the backdoor link, but without enabling OSPF sham-link:

Traceroute from DL2 to DL6 shows traffic flowing over the backdoor link:

```
DL2-DC2-RED#traceroute vrf red-data 1.1.1.16
Type escape sequence to abort.
Tracing the route to 1.1.1.16
  1 125.1.99.2 0 msec 0 msec 4 msec
  2 125.1.99.5 0 msec *  0 msec
DL2-DC2-RED#
```

Traceroute from DL6 to DL2 shows traffic flowing over the backdoor link:

```
DL6-MC-RED#traceroute 1.1.1.23

Type escape sequence to abort.
Tracing the route to 1.1.1.23

  1 125.1.99.6 0 msec 0 msec 0 msec
  2 125.1.99.1 0 msec *  0 msec
DL6-MC-RED#
```

Traceroute from DL2 to DL1m shows traffic flowing over the MPLS VPN network:

```
DL2-DC2-RED#traceroute vrf red-data 1.1.1.1
Type escape sequence to abort.
Tracing the route to 1.1.1.1

  1 125.1.102.49 0 msec
    125.1.102.33 0 msec
    125.1.102.49 0 msec
  2 125.1.102.13 [MPLS: Labels 26/68 Exp 0] 0 msec
    125.1.102.17 [MPLS: Labels 26/68 Exp 0] 0 msec
    125.1.102.13 [MPLS: Labels 26/68 Exp 0] 0 msec
  3 125.1.101.9 [MPLS: Label 68 Exp 0] 4 msec 0 msec 0 msec
4 125.1.101.10 0 msec *  0 msec
```

Traceroute from DL1 to DL6 shows traffic flowing over the MPLS VPN network:

```
DL1-DC1-RED# traceroute vrf red-data 1.1.1.16

Type escape sequence to abort.
Tracing the route to 1.1.1.16

  1 125.1.101.1 0 msec
    125.1.101.9 0 msec
    125.1.101.1 4 msec
  2 125.1.100.69 [MPLS: Labels 59/73 Exp 0] 0 msec
    125.1.100.61 [MPLS: Labels 63/76 Exp 0] 0 msec
    125.1.100.69 [MPLS: Labels 59/73 Exp 0] 0 msec
  3 125.1.104.1 [MPLS: Label 76 Exp 0] 0 msec
    125.1.104.9 [MPLS: Label 73 Exp 0] 0 msec
    125.1.104.1 [MPLS: Label 76 Exp 0] 4 msec
  4 125.1.104.10 0 msec
    125.1.104.2 0 msec *
```

Traceroute from the backdoor link router to DL1, DL2 and DL6:

```
3600-bd-red#traceroute 1.1.1.1

Type escape sequence to abort.
Tracing the route to 1.1.1.1

  1 125.1.99.5 0 msec
    125.1.99.1 0 msec
    125.1.99.5 0 msec
  2 125.1.102.49 4 msec
    125.1.104.9 0 msec
    125.1.102.49 0 msec
  3 125.1.100.37 [MPLS: Labels 47/73 Exp 0] 4 msec
    125.1.102.17 [MPLS: Labels 26/68 Exp 0] 0 msec
    125.1.100.37 [MPLS: Labels 47/73 Exp 0] 4 msec
  4 125.1.101.9 [MPLS: Label 68 Exp 0] 0 msec
    125.1.100.74 [MPLS: Labels 23/73 Exp 0] 0 msec
    125.1.101.9 [MPLS: Label 68 Exp 0] 0 msec
  5 125.1.101.1 [MPLS: Label 73 Exp 0] 0 msec
    125.1.101.10 4 msec
    125.1.101.1 [MPLS: Label 73 Exp 0] 0 msec

3600-bd-red#traceroute 1.1.1.23

Type escape sequence to abort.
Tracing the route to 1.1.1.23

  1 125.1.99.1 0 msec *  0 msec

3600-bd-red#traceroute 1.1.1.16

Type escape sequence to abort.
Tracing the route to 1.1.1.16

  1 125.1.99.5 0 msec *  0 msec
3600-bd-red#
```

With the backdoor link between Data Center 2 (DL2) and Medium Campus (DL6), but OSPF sham-link configured on the relevant PEs; in this case, between PE3, PE4 and PE7, PE8 pairs:

Traceroute from DL2 to DL6 shows traffic flowing over the MPLS network:

```
DL2-DC2-RED#traceroute vrf red-data 1.1.1.16

Type escape sequence to abort.
Tracing the route to 1.1.1.16

  1 125.1.102.33 0 msec
    125.1.102.49 0 msec
    125.1.102.33 0 msec
  2 125.1.102.5 [MPLS: Labels 63/76 Exp 0] 0 msec
    125.1.102.1 [MPLS: Labels 63/76 Exp 0] 0 msec
    125.1.102.5 [MPLS: Labels 63/76 Exp 0] 0 msec
  3 125.1.104.1 [MPLS: Label 76 Exp 0] 4 msec 0 msec 0 msec
  4 125.1.104.2 0 msec *  0 msec
```

Traceroute from DL6 to DL2 shows traffic flowing over the MPLS VPN network:

```
DL6-MC-RED#traceroute 1.1.1.23

Type escape sequence to abort.
Tracing the route to 1.1.1.23
```

```
1 125.1.104.1 0 msec
  125.1.104.9 0 msec
  125.1.104.1 0 msec
2 125.1.100.37 [MPLS: Labels 50/88 Exp 0] 4 msec
  125.1.100.33 [MPLS: Labels 37/88 Exp 0] 0 msec
  125.1.100.37 [MPLS: Labels 50/88 Exp 0] 0 msec
3 125.1.102.33 [MPLS: Label 88 Exp 0] 0 msec 0 msec 0 msec
4 125.1.102.34 0 msec *  0 msec
```

QoS

Because only queueing was implemented on the Cisco 7600 PEs, the verification involves checking to see whether DSCP is being trusted as configured and whether priority for real-time traffic is maintained in case of congestion.

Oversubscribing the PE to CE egress link:

```
7600-DC2-PE3#sh queueing int g1/4
Interface GigabitEthernet1/4 queueing strategy:  Weighted Round-Robin
  Port QoS is enabled
  Trust state: trust DSCP
  Extend trust state: not trusted [COS = 0]
  Default COS is 0
    Queueing Mode In Tx direction: mode-cos
    Transmit queues [type = 1p3q8t]:
    Queue Id    Scheduling  Num of thresholds
    -----------------------------------------
       01         WRR                 08
       02         WRR                 08
       03         WRR                 08
       04         Priority            01

    WRR bandwidth ratios:    5[queue 1]  25[queue 2]  70[queue 3]
    queue-limit ratios:      5[queue 1]  25[queue 2]  40[queue 3]

    queue tail-drop-thresholds
    --------------------------
    1     70[1] 100[2] 100[3] 100[4] 100[5] 100[6] 100[7] 100[8]
    2     70[1] 100[2] 100[3] 100[4] 100[5] 100[6] 100[7] 100[8]
    3     100[1] 100[2] 100[3] 100[4] 100[5] 100[6] 100[7] 100[8]

    queue random-detect-min-thresholds
    ----------------------------------
      1    80[1] 100[2] 100[3] 100[4] 100[5] 100[6] 100[7] 100[8]
      2    80[1] 100[2] 100[3] 100[4] 100[5] 100[6] 100[7] 100[8]
      3    50[1] 60[2] 70[3] 80[4] 90[5] 100[6] 100[7] 100[8]

    queue random-detect-max-thresholds
    ----------------------------------
      1    100[1] 100[2] 100[3] 100[4] 100[5] 100[6] 100[7] 100[8]
      2    100[1] 100[2] 100[3] 100[4] 100[5] 100[6] 100[7] 100[8]
      3    60[1] 70[2] 80[3] 90[4] 100[5] 100[6] 100[7] 100[8]

    WRED disabled queues:

    queue thresh cos-map
    ---------------------------------------
    1     1      1
    1     2
    1     3
    1     4
```

```
1     5
1     6
1     7
1     8
2     1       0
2     2
2     3
2     4
2     5
2     6
2     7
2     8
3     1       4
3     2       2
3     3       3
3     4       6
3     5       7
3     6
3     7
3     8
4     1       5

  Queueing Mode In Rx direction: mode-cos
  Receive queues [type = 2q8t]:
  Queue Id    Scheduling  Num of thresholds
  -----------------------------------------
     01         WRR                 08
     02         WRR                 08

  WRR bandwidth ratios:  100[queue 1]   0[queue 2]
  queue-limit ratios:    100[queue 1]   0[queue 2]

  queue tail-drop-thresholds
  --------------------------
  1     100[1] 100[2] 100[3] 100[4] 100[5] 100[6] 100[7] 100[8]
  2     100[1] 100[2] 100[3] 100[4] 100[5] 100[6] 100[7] 100[8]

  queue thresh cos-map
  ---------------------------------------
  1     1       0 1 2 3 4 5 6 7
  1     2
  1     3
  1     4
  1     5
  1     6
  1     7
  1     8
  2     1
  2     2
  2     3
  2     4
  2     5
  2     6
  2     7
  2     8


Packets dropped on Transmit:

  queue     dropped  [cos-map]
  ---------------------------------------------
  1            11686  [1 ]
  2                0  [0 ]
  3                0  [4 2 3 6 7 ]
```

```
    4                0  [5 ]

  Packets dropped on Receive:

    queue     dropped  [cos-map]
    ---------------------------------------------
    1                0  [0 1 2 3 4 5 6 7 ]
    2                0  []
```

Multicast

The multicast service can be verified by performing the following steps. PE3 is the PE closest to the source, and PE8 is the PE closest to the test receiver.

1. Verify the BGP updates:

```
        sh ip pim mdt bgp

7600-DC2-PE3#sh ip pim mdt bgp
Peer (Route Distinguisher + IPv4)                                    Next Hop
  MDT group 239.232.10.4
   2:10:1041:125.1.125.5                                             125.1.125.5
   2:10:1042:125.1.125.6                                             125.1.125.6
   2:10:1044:125.1.125.8                                             125.1.125.8
  MDT group 239.232.10.1
   2:10:1054:125.1.125.8                                             125.1.125.8
   2:10:1055:125.1.125.9                                             125.1.125.9
   2:10:1056:125.1.125.10                                            125.1.125.10
   2:10:105:125.1.125.13                                             125.1.125.13
  MDT group 239.232.10.3
   2:10:1031:125.1.125.5                                             125.1.125.5
   2:10:1032:125.1.125.6                                             125.1.125.6
   2:10:1034:125.1.125.8                                             125.1.125.8
   2:10:1037:125.1.125.11                                            125.1.125.11
   2:10:1038:125.1.125.12                                            125.1.125.12
  MDT group 239.232.10.2
   2:10:1064:125.1.125.8                                             125.1.125.8
   2:10:1065:125.1.125.9                                             125.1.125.9
   2:10:1066:125.1.125.10                                            125.1.125.10
   2:10:106:125.1.125.13                                             125.1.125.13

        sh ip bgp vpnv4 all

7600-DC2-PE3#sh ip bgp vpnv4 all
BGP table version is 32649, local router ID is 125.1.125.7
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path

<snip>

Route Distinguisher: 2:10:105
* i125.1.125.13/32  125.1.125.13             0    100      0 ?
*>i                 125.1.125.13             0    100      0 ?
Route Distinguisher: 2:10:106
* i125.1.125.13/32  125.1.125.13             0    100      0 ?
*>i                 125.1.125.13             0    100      0 ?
Route Distinguisher: 2:10:1031
* i125.1.125.5/32   125.1.125.5              0    100      0 ?
```

```
*>i                 125.1.125.5              0    100      0 ?
Route Distinguisher: 2:10:1032
* i125.1.125.6/32   125.1.125.6              0    100      0 ?
*>i                 125.1.125.6              0    100      0 ?
Route Distinguisher: 2:10:1033
*> 125.1.125.7/32   0.0.0.0                                0 ?
Route Distinguisher: 2:10:1034
* i125.1.125.8/32   125.1.125.8              0    100      0 ?
*>i                 125.1.125.8              0    100      0 ?
Route Distinguisher: 2:10:1037
* i125.1.125.11/32  125.1.125.11             0    100      0 ?
*>i                 125.1.125.11             0    100      0 ?
Route Distinguisher: 2:10:1038
* i125.1.125.12/32  125.1.125.12             0    100      0 ?
*>i                 125.1.125.12             0    100      0 ?
Route Distinguisher: 2:10:1041
* i125.1.125.5/32   125.1.125.5              0    100      0 ?
*>i                 125.1.125.5              0    100      0 ?
Route Distinguisher: 2:10:1042
* i125.1.125.6/32   125.1.125.6              0    100      0 ?
*>i                 125.1.125.6              0    100      0 ?
Route Distinguisher: 2:10:1043
*> 125.1.125.7/32   0.0.0.0                                0 ?
Route Distinguisher: 2:10:1044
* i125.1.125.8/32   125.1.125.8              0    100      0 ?
*>i                 125.1.125.8              0    100      0 ?
Route Distinguisher: 2:10:1053
*> 125.1.125.7/32   0.0.0.0                                0 ?
Route Distinguisher: 2:10:1054
   Network          Next Hop            Metric LocPrf Weight Path
* i125.1.125.8/32   125.1.125.8              0    100      0 ?
*>i                 125.1.125.8              0    100      0 ?
Route Distinguisher: 2:10:1055
* i125.1.125.9/32   125.1.125.9              0    100      0 ?
*>i                 125.1.125.9              0    100      0 ?
Route Distinguisher: 2:10:1056
* i125.1.125.10/32  125.1.125.10             0    100      0 ?
*>i                 125.1.125.10             0    100      0 ?
Route Distinguisher: 2:10:1063
*> 125.1.125.7/32   0.0.0.0                                0 ?
Route Distinguisher: 2:10:1064
* i125.1.125.8/32   125.1.125.8              0    100      0 ?
*>i                 125.1.125.8              0    100      0 ?
Route Distinguisher: 2:10:1065
* i125.1.125.9/32   125.1.125.9              0    100      0 ?
*>i                 125.1.125.9              0    100      0 ?
Route Distinguisher: 2:10:1066
* i125.1.125.10/32  125.1.125.10             0    100      0 ?
*>i                 125.1.125.10             0    100      0 ?
```

2. Verify the global mroute table:

```
sh ip mroute <>

7600-DC2-PE3# sh ip mroute
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report, Z - Multicast Tunnel
       Y - Joined MDT-data group, y - Sending to MDT-data group
Outgoing interface flags: H - Hardware switched, A - Assert winner
```

```
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(125.1.125.7, 239.232.10.1), 3d14h/00:03:21, flags: sTZ
  Incoming interface: Loopback0, RPF nbr 0.0.0.0, RPF-MFD
  Outgoing interface list:
    GigabitEthernet1/2, Forward/Sparse, 3d14h/00:02:37, H
    GigabitEthernet1/3, Forward/Sparse, 3d14h/00:02:49, H

(125.1.125.8, 239.232.10.1), 3d14h/00:02:51, flags: sTIZ
  Incoming interface: GigabitEthernet1/3, RPF nbr 125.1.102.22, RPF-MFD
  Outgoing interface list:
    MVRF blue-data, Forward/Sparse, 3d14h/00:02:40, H

(125.1.125.9, 239.232.10.1), 3d14h/00:02:51, flags: sTIZ
  Incoming interface: GigabitEthernet1/2, RPF nbr 125.1.102.13, RPF-MFD
  Outgoing interface list:
    MVRF blue-data, Forward/Sparse, 3d14h/00:00:50, H

(125.1.125.10, 239.232.10.1), 1d04h/00:02:51, flags: sTIZ
  Incoming interface: GigabitEthernet1/2, RPF nbr 125.1.102.13, RPF-MFD
  Outgoing interface list:
    MVRF blue-data, Forward/Sparse, 05:00:10/00:02:55, H

(125.1.125.13, 239.232.10.1), 3d14h/00:02:51, flags: sTIZ
  Incoming interface: GigabitEthernet1/1, RPF nbr 125.1.102.1, RPF-MFD
  Outgoing interface list:
    MVRF blue-data, Forward/Sparse, 3d14h/00:02:40, H

(125.1.125.7, 239.232.10.2), 3d14h/00:03:21, flags: sTZ
  Incoming interface: Loopback0, RPF nbr 0.0.0.0, RPF-MFD
  Outgoing interface list:
    GigabitEthernet1/2, Forward/Sparse, 3d14h/00:02:49, H
    GigabitEthernet1/3, Forward/Sparse, 3d14h/00:02:50, H

(125.1.125.8, 239.232.10.2), 3d14h/00:02:51, flags: sTIZ
  Incoming interface: GigabitEthernet1/3, RPF nbr 125.1.102.22, RPF-MFD
  Outgoing interface list:
    MVRF blue-voice, Forward/Sparse, 3d14h/00:02:40, H

(125.1.125.9, 239.232.10.2), 3d14h/00:02:51, flags: sTIZ
  Incoming interface: GigabitEthernet1/2, RPF nbr 125.1.102.13, RPF-MFD
  Outgoing interface list:
    MVRF blue-voice, Forward/Sparse, 3d14h/00:00:50, H

(125.1.125.10, 239.232.10.2), 1d04h/00:02:51, flags: sTIZ
  Incoming interface: GigabitEthernet1/2, RPF nbr 125.1.102.13, RPF-MFD
  Outgoing interface list:
    MVRF blue-voice, Forward/Sparse, 05:00:07/00:02:58, H

(125.1.125.13, 239.232.10.2), 3d14h/00:02:50, flags: sTIZ
  Incoming interface: GigabitEthernet1/1, RPF nbr 125.1.102.1, RPF-MFD
  Outgoing interface list:
    MVRF blue-voice, Forward/Sparse, 3d14h/00:02:39, H

(125.1.125.5, 239.232.10.3), 3d14h/00:02:50, flags: sTIZ
  Incoming interface: GigabitEthernet1/1, RPF nbr 125.1.102.1, RPF-MFD
  Outgoing interface list:
    MVRF red-data, Forward/Sparse, 3d14h/00:00:49, H

(125.1.125.6, 239.232.10.3), 3d14h/00:02:50, flags: sTIZ
  Incoming interface: GigabitEthernet1/2, RPF nbr 125.1.102.13, RPF-MFD
  Outgoing interface list:
    MVRF red-data, Forward/Sparse, 3d14h/00:00:49, H
```

```
(125.1.125.7, 239.232.10.3), 3d14h/00:03:20, flags: sTZ
  Incoming interface: Loopback0, RPF nbr 0.0.0.0, RPF-MFD
  Outgoing interface list:
    GigabitEthernet1/2, Forward/Sparse, 3d14h/00:02:44, H
    GigabitEthernet1/1, Forward/Sparse, 3d14h/00:02:45, H
    GigabitEthernet1/3, Forward/Sparse, 3d14h/00:03:04, H

(125.1.125.8, 239.232.10.3), 3d14h/00:02:50, flags: sTIZ
  Incoming interface: GigabitEthernet1/3, RPF nbr 125.1.102.22, RPF-MFD
  Outgoing interface list:
    MVRF red-data, Forward/Sparse, 3d14h/00:02:39, H

(125.1.125.11, 239.232.10.3), 3d14h/00:02:50, flags: sTIZ
  Incoming interface: GigabitEthernet1/1, RPF nbr 125.1.102.1, RPF-MFD
  Outgoing interface list:
    MVRF red-data, Forward/Sparse, 3d14h/00:01:20, H

(125.1.125.12, 239.232.10.3), 3d14h/00:02:50, flags: sTIZ
  Incoming interface: GigabitEthernet1/2, RPF nbr 125.1.102.13, RPF-MFD
  Outgoing interface list:
    MVRF red-data, Forward/Sparse, 3d14h/00:00:49, H

(125.1.125.5, 239.232.10.4), 3d14h/00:02:50, flags: sTIZ
  Incoming interface: GigabitEthernet1/1, RPF nbr 125.1.102.1, RPF-MFD
  Outgoing interface list:
    MVRF red-voice, Forward/Sparse, 3d14h/00:00:49, H

(125.1.125.6, 239.232.10.4), 3d14h/00:02:50, flags: sTIZ
  Incoming interface: GigabitEthernet1/2, RPF nbr 125.1.102.13, RPF-MFD
  Outgoing interface list:
    MVRF red-voice, Forward/Sparse, 3d14h/00:00:49, H

(125.1.125.7, 239.232.10.4), 3d14h/00:03:27, flags: sTZ
  Incoming interface: Loopback0, RPF nbr 0.0.0.0, RPF-MFD
  Outgoing interface list:
    GigabitEthernet1/2, Forward/Sparse, 3d14h/00:03:27, H
    GigabitEthernet1/1, Forward/Sparse, 3d14h/00:03:02, H
    GigabitEthernet1/3, Forward/Sparse, 3d14h/00:03:10, H

(125.1.125.8, 239.232.10.4), 3d14h/00:02:50, flags: sTIZ
  Incoming interface: GigabitEthernet1/3, RPF nbr 125.1.102.22, RPF-MFD
  Outgoing interface list:
    MVRF red-voice, Forward/Sparse, 3d14h/00:02:38, H

(125.1.125.7, 239.232.20.32), 00:02:29/00:03:20, flags: sT
  Incoming interface: Loopback0, RPF nbr 0.0.0.0, RPF-MFD
  Outgoing interface list:
    GigabitEthernet1/2, Forward/Sparse, 00:02:29/00:02:56, H

(125.1.125.8, 239.232.20.32), 00:03:47/00:02:12, flags: sPT
  Incoming interface: GigabitEthernet1/3, RPF nbr 125.1.102.22, RPF-MFD
  Outgoing interface list: Null

(*, 224.0.1.40), 3d14h/00:02:09, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Loopback0, Forward/Sparse, 3d14h/00:02:09
```

3. Verify the mroutes in the VRF (on the receiving router):

```
       7600-MC-PE8#sh ip mroute vrf red-data
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
```

```
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report, Z - Multicast Tunnel
       Y - Joined MDT-data group, y - Sending to MDT-data group
Outgoing interface flags: H - Hardware switched, A - Assert winner
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.232.10.1), 02:57:22/00:03:13, RP 3.3.3.11, flags: S
  Incoming interface: Tunnel0, RPF nbr 125.1.125.7, RPF-MFD
  Outgoing interface list:
    GigabitEthernet1/3, Forward/Sparse, 02:57:22/00:03:13, H

(125.1.2.66, 224.232.10.1), 02:57:22/00:03:23, flags: TY
  Incoming interface: Tunnel0, RPF nbr 125.1.125.7, RPF-MFD, MDT:[125.1.125.7,23
9.232.20.32]/00:02:44
  Outgoing interface list:
    GigabitEthernet1/3, Forward/Sparse, 02:57:22/00:03:13, H

(*, 224.0.1.40), 3d17h/00:03:10, RP 3.3.3.11, flags: SJCL
  Incoming interface: Tunnel0, RPF nbr 125.1.125.7
  Outgoing interface list:
    GigabitEthernet1/3, Forward/Sparse, 3d17h/00:03:10

(*, 239.255.255.250), 02:57:22/00:03:02, RP 3.3.3.11, flags: S
  Incoming interface: Tunnel0, RPF nbr 125.1.125.7, RPF-MFD
  Outgoing interface list:
    GigabitEthernet1/3, Forward/Sparse, 02:57:22/00:03:02, H
```

4. Verify the PIM neighbors in the global table:

```
sh ip pim nei

7600-DC2-PE3#sh ip pim nei
PIM Neighbor Table
Neighbor          Interface               Uptime/Expires    Ver   DR
Address                                                           Prio/Mode
125.1.102.1       GigabitEthernet1/1      3d14h/00:01:17    v2    1 /
125.1.102.13      GigabitEthernet1/2      3d14h/00:01:23    v2    1 / S
125.1.102.22      GigabitEthernet1/3      3d14h/00:01:26    v2    1 / DR
```

5. Verify the PIM neighbors with the VPN

```
   sh ip pim vrf <> nei

7600-DC2-PE3#sh ip pim vrf red-data nei
PIM Neighbor Table
Neighbor          Interface               Uptime/Expires    Ver   DR
Address                                                           Prio/Mode
125.1.102.34      GigabitEthernet1/4.1    3d14h/00:01:30    v2    1 / DR S
125.1.125.11      Tunnel1                 3d14h/00:01:17    v2    1 / S
125.1.125.5       Tunnel1                 3d14h/00:01:16    v2    1 / S
125.1.125.6       Tunnel1                 3d14h/00:01:37    v2    1 / S
125.1.125.12      Tunnel1                 3d14h/00:01:34    v2    1 / DR S
125.1.125.8       Tunnel1                 3d14h/00:01:17    v2    1 /
```

APPENDIX A

Platform-Specific Capabilities and Constraints

Platforms within the NG-WAN/MAN architecture perform QoS either in software or hardware. Furthermore, hardware QoS is hardware-specific and thus may vary significantly platform-to-platform, even from line card-to-line card. QoS functionality may vary in subtle ways, as may QoS command syntax. The following sections address the main platform-specific concerns and design recommendations for the NG-WAN/MAN. The platforms discussed include:

- Cisco 7200
- Cisco 7304
- Cisco 7600
- Cisco 12000 Gigabit Switch Router (GSR)

Cisco 7200 QoS Design

As previously mentioned, the Cisco 7200 series router performs QoS within Cisco IOS software. However because QoS is performed in software, QoS policies require marginal CPU processing to implement. The degree of impact is a function of the complexity of the policy, the traffic profile, the speeds, and the Network Processing Engine hardware. The rule of thumb to keep in mind is to design and test QoS policies such that when enabled the CPU does not exceed 75 percent utilization during normal operation. The first section examines configuring Uniform Mode MPLS DiffServ Tunneling on these platforms and the following sections examine designs for an 8-class QoS model and an 11-class QoS for these platforms.

Cisco 7200—Uniform Mode MPLS DiffServ Tunneling

Configuring uniform mode MPLS DiffServ Tunneling on the Cisco 7200 requires two parts, but the first is by default. Specifically, the mapping of IP Precedence to MPLS EXP is performed on Cisco 7200 PE routers (for customer-to-provider traffic) by default.

However for PE-to-CE egress traffic (exiting the MPLS VPN), additional configuration is required on the PE to achieve mapping of MPLS EXP to IP Precedence. This is because the final label is popped (and discarded) when it is received from the MPLS VPN cloud and therefore cannot be used as a match criterion for policies applied to the egress interface of the final PE router (facing the destination CE). The solution is to copy the final MPLS EXP bit values to a temporary placeholder on the PE ingress from the MPLS core (before the label is discarded) and then use these temporary placeholder values for setting the IP Precedence bits on egress to the CE.

Cisco IOS provides two such temporary placeholders, the QoS group and the discard class. For uniform mode scenarios, it is recommended to copy the MPLS EXP values to QoS group values on the ingress from the MPLS VPN cloud. (The discard class is recommended for use in pipe mode scenarios only.) QoS group values can then be copied to IP Precedence values (on egress to the customer CE).

The following is a sample Cisco 7200 uniform mode configuration.

```
!
policy-map MPLSEXP-TO-QOSGROUP
 class class-default
  set qos-group mpls experimental topmost           ! Copies EXP to QoS Group
!
policy-map QOSGROUP-TO-IPP
 class class-default
  set precedence qos-group           ! Copies QoS Group to IPP
!
!
interface GigabitEthernet1/0
 description GE TO MPLS VPN CORE         ! Link to/from MPLS VPN Core
 ip address 20.2.34.4 255.255.255.0
 ip vrf forwarding RED
 ip address 10.1.45.4 255.255.255.0
 service-policy input MPLSEXP-TO-QOSGROUP           ! MPLS EXP to QoS Group
 tag-switching ip
!
…
!
interface FastEthernet2/0
 description GE TO RED CE         ! Link to/from CE
 ip vrf forwarding RED
 ip address 10.1.45.4 255.255.255.0
 service-policy output QOSGROUP-TO-IPP           ! QoS Group to IPP
```

Cisco 7200—8-Class QoS Model

In the 8-class model is provisioned for the following application types:

- Voice
- Interactive-Video (video-conferencing)
- Network control (a combination of IP Routing and Network Management traffic)
- Call-Signaling
- Critical Data
- Bulk Data
- Best Effort
- Scavenger

This model takes advantage of the implicit policer function with Cisco IOS LLQ that allows you to time-division multiplex the LLQ. Essentially this functionality allows you to configure “dual-LLQs” even though only a single LLQ is operational. For example, assume you have configured one LLQ for Voice, set to 100 kbps, and another LLQ for Interactive-Video, set to 400 kbps. The software actually provisions a single LLQ for 500 kbps and allows only up to 100 kbps of Voice traffic and 400 kbps worth of Interactive-Video traffic into this queue on a first-in, first-out (FIFO) basis. If more than 100 kbps of voice is offered to this LLQ, it is dropped, and if more than 400 kbps of Interactive-Video is offered to it, it is also dropped. In this manner, both Voice and Interactive-Video benefit from Strict Priority

servicing and, at the same time, data applications are protected from starvation (via the implicit policer). Following the LLQ queuing best-practice design principle presented previously in this chapter, Cisco recommends that the sum of the LLQs be less than 33 percent of a given link.

Note This implicit policing functionality of Cisco IOS LLQ should further impress on the network administrator the need to accurately provision Call Admission Control (CAC) to be in sync with the LLQ-provisioned bandwidth.

Voice is marked as EF, which is set by default on Cisco IP phones. When identified, VoIP is admitted into its own LLQ which, in this example, is set to 18 percent. CAC correspondingly should be assigned to this link by dividing the allocated bandwidth by the voice codec (including Layer 2 overhead) to determine how many calls can be permitted simultaneously over this link.

Interactive-Video (also known as IP videoconferencing [IP/VC]) is recommended to be marked AF41 (which can be marked down to AF42 or AF43, in the case of single- or dual-rate policing at the campus access edge). Interactive-Video is also assigned an LLQ under this dual-LLQ design. Cisco recommends overprovisioning the Interactive-Video LLQ by 20 percent of the IP/VC rate. This takes into account IP/UDP/RTP headers as well as Layer 2 overhead.

Additionally, Cisco IOS Software automatically includes a 200-ms burst parameter (defined in bytes) as part of the priority command. This default burst parameter has tested sufficient for protecting a single 384-kbps IP/VC stream; on higher speed links, the default burst parameter has shown to be insufficient for protecting multiple IP/VC streams. However multiple-stream IP/VC quality tested well with the burst set to 30,000 bytes (for example, priority 920 30000). The main point is that the default LLQ burst parameter might require tuning as multiple IP/VC streams are added.

Optionally DSCP-based WRED can be enabled on the Interactive-Video class, but testing has shown negligible performance difference in doing so. This is because congestion avoidance algorithms such as WRED are more effective on TCP-based flows than UDP-based flows, such as Interactive-Video.

A Network Control class is included within this 8-class model to protect network control plane traffic, specifically IP Routing (marked as CS6) and Network Management traffic (marked as CS2). As previously mentioned, Interior Gateway Protocol packets (such as RIP, EIGRP, OSPF, and IS-IS) are protected through the PAK_priority mechanism within the router. However EGP protocols such as BGP do not get PAK_priority treatment and might need explicit bandwidth guarantees to ensure that peering sessions do not reset during periods of congestion. Additionally administrators might want to protect network management access to devices during periods of congestion.

Call-Signaling traffic is also marked on the IP phones as CS3 (although some older versions of Cisco CallManager may mark Call-Signaling to the legacy value of AF31); Call-Signaling requires a moderate but dedicated bandwidth guarantee.

In these designs, WRED is not enabled on classes such as Network Control (IP Routing/Network Management) and Call-Signaling, because WRED would take effect only if such classes were filling their queues nearly to their limits. Such conditions would indicate an under-provisioning problem that would be better addressed by increasing the minimum bandwidth allocation for these classes rather than by enabling WRED.

The Critical Data class requires Transactional/Interactive Data traffic to be marked to AF21 (or AF22 or AF23 in the case of single- or dual-rate policers deployed within the campus). A bandwidth guarantee is made for this class, and DSCP-based WRED is enabled on this class to achieve the RFC 2597 Assured Forwarding Per-Hop Behavior.

The Bulk Data class requires Bulk Data to be marked to AF11 (or AF12 or AF13 in the case of single- or dual-rate policing deployed within the campus). Because TCP continually increases its window sizes, which is especially noticeable in long sessions such as large file transfers, constraining Bulk Data to its

own class alleviates other data classes from being dominated by such large file transfers. A moderate bandwidth guarantee is made for this class and DSCP-based WRED is enabled on this class to achieve the RFC 2597 Assured Forwarding Per-Hop Behavior.

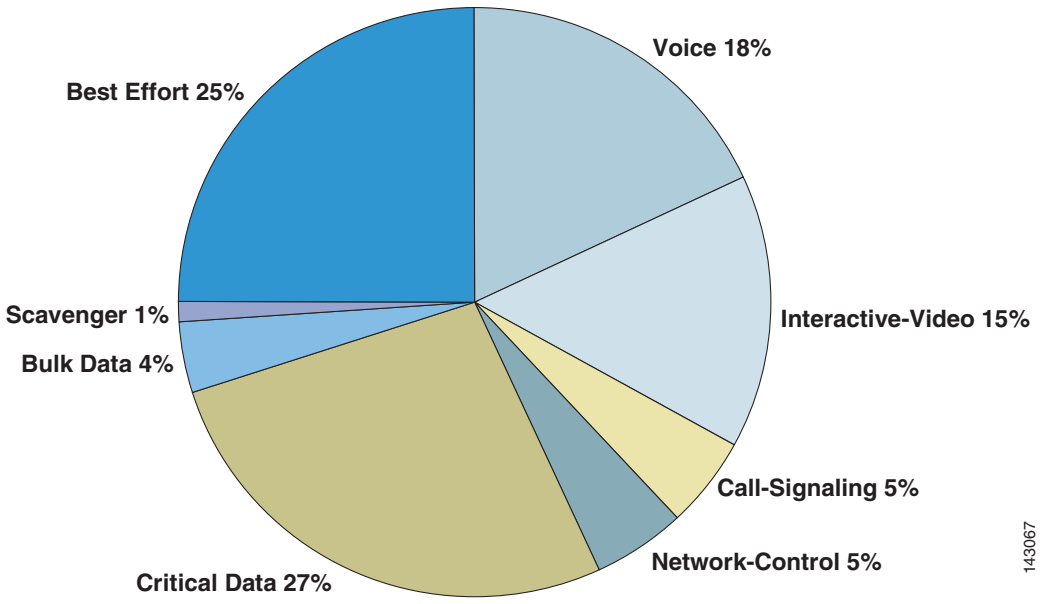
The Scavenger class constrains any traffic marked to DSCP CS1 to 1 percent of the link during periods of congestion; this allows class-default to use the remaining 25 percent. However to constrain Scavenger to 1 percent, an explicit bandwidth guarantee of 25 percent must be given to the Best Effort class. Otherwise if class-default is not explicitly assigned a minimum bandwidth guarantee, the Scavenger class can still rob it of bandwidth. This is because of the way the CBWFQ algorithm has been coded; if classes protected with a bandwidth statement are offered more traffic than their minimum bandwidth guarantee, the algorithm tries to protect such excess traffic at the direct expense of robbing bandwidth from class-default (if class-default is configured with fair-queue), unless class-default itself has a bandwidth statement that provides itself with a minimum bandwidth guarantee. However assigning a bandwidth statement to class-default on non-distributed platforms such as the Cisco 7200 currently precludes the enabling of fair queuing (fair-queue) on this class and forces FIFO queuing on class-default.

An additional implication of using a bandwidth statement on class-default is that even though 25 percent of the link is reserved explicitly for class-default, the parser does not attach the policy to a physical interface unless the **max-reserved-bandwidth 100** command is entered on the interface before the service-policy output statement. This is because the parser adds the sum of the bandwidth statements (regardless of whether one of these is applied to the class-default) and if the total is in excess of 75 percent of the link bandwidth, rejects the application of the policy to the interface.

Finally WRED can be enabled on the Best Effort class to provide congestion management on this default-class. Because all traffic assigned to the default class is to be marked to the same DSCP value of 0, it is superfluous to enable DSCP-based WRED on such a class; WRED (technically RED in this case because all the IP Precedence weights are the same) would suffice.

The Cisco 7200 8-Class QoS Model is shown in Figure A-1.

Figure A-1 Cisco 7200 8-Class QoS Model Example



The following configuration example shows the corresponding configuration for this Cisco 7200 8-Class Model. Keep in mind that this model is intended for links of speeds greater than 3 Mbps.

```
!
class-map match-all VOICE
 match ip dscp ef                 ! QoS Baseline marking for Voice
class-map match-all INTERACTIVE-VIDEO
 match ip dscp af41 af42 af43         ! QoS Baseline marking Interactive-Video
class-map match-any NETWORK-CONTROL
 match ip dscp cs6                ! QoS Baseline marking for IP Routing
 match ip dscp cs2                ! QoS Baseline marking for Network Mgmt
class-map match-all CALL-SIGNALING
 match ip dscp cs3                ! QoS Baseline marking for Call-Signaling
class-map match-all CRITICAL-DATA
 match ip dscp af21 af22 af23         ! QoS Baseline marking Transactional-Data
class-map match-all BULK-DATA
 match ip dscp af11 af12 af13         ! QoS Baseline marking for Bulk-Data
class-map match-all SCAVENGER
 match ip dscp cs1                ! QoS Baseline marking for Scavenger
!
policy-map WAN-EDGE
 class VOICE
  priority percent 18                 ! Voice gets LLQ - "dual-LLQ" design
 class INTERACTIVE-VIDEO
  priority percent 15                 ! Int-Video gets LLQ - "dual-LLQ" design
 class NETWORK-CONTROL
  bandwidth percent 5                 ! Routing and Network Mgmt gets CBWFQ
 class CALL-SIGNALING
  bandwidth percent 5                 ! Call-Signaling gets CBWFQ
 class CRITICAL-DATA
  bandwidth percent 27                ! Critical Data gets CBWFQ
  random-detect dscp-based            ! Critical Data also gets DSCP-based WRED
 class BULK-DATA
  bandwidth percent 4                 ! Bulk Data gets CBWFQ
  random-detect dscp-based            ! Bulk Data also gets DSCP-based WRED
 class SCAVENGER
  bandwidth percent 1                 ! Scavenger gets minimum CBWFQ
 class class-default
  bandwidth percent 25                ! Best Effort is protected with CBWFQ
  random-detect                    ! Best Effort also gets WRED (RED)
!
```

Cisco 7200—11-Class QoS Model

As mentioned previously, the 11-class QoS Baseline is a guiding model for addressing the QoS needs of today and the foreseeable future. The QoS Baseline is not a mandate dictating what enterprises must deploy today; instead this strategic document offers standards-based recommendations for marking and provisioning traffic classes that allow for greater interoperability and simplified future expansion.

Building on the previous 8-class model and as illustrated in Figure A-1, the Network Control class is subdivided into the IP Routing and Network Management classes.

Additionally the Critical Data class is subdivided into the Mission-Critical Data and Transactional Data classes.

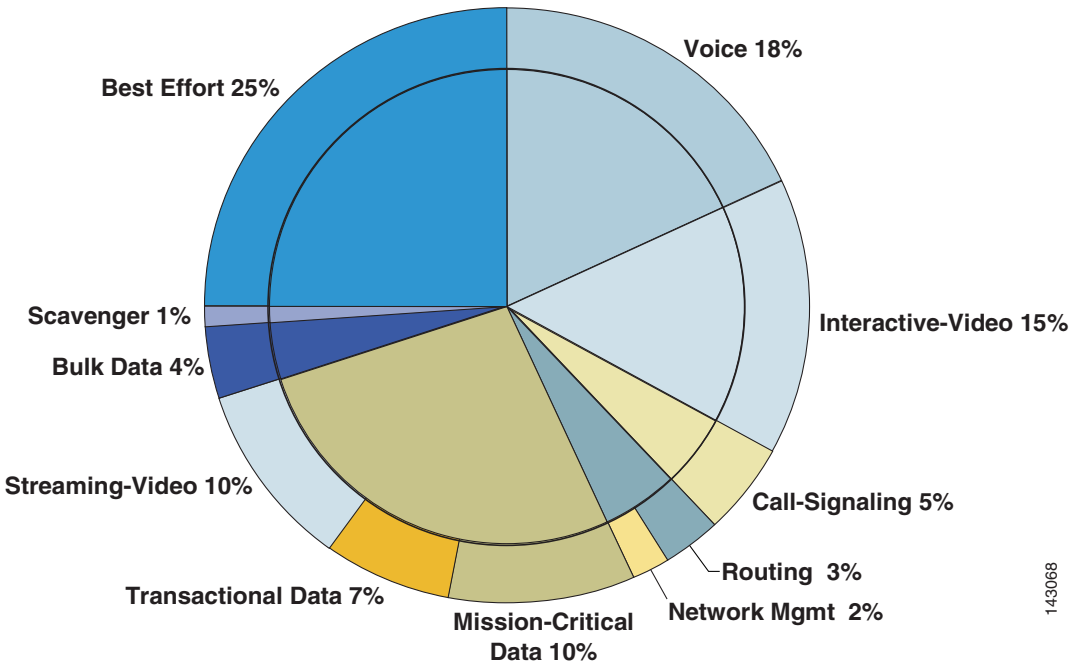
The Locally-Defined Mission-Critical Data class requires Mission-Critical Data traffic to be marked to AF31 (or AF32 or AF33 in the case of single- or dual-rate policers deployed within the campus). A bandwidth guarantee is made for this class and DSCP-based WRED is enabled on this class to achieve the RFC 2597 Assured Forwarding Per-Hop Behavior.

Note This recommendation assumes that Call-Signaling migration from AF31 to CS3 is complete within the enterprise; if not, then a temporary, non-standard DSCP value, such as 25, might be used for Mission-Critical Data marking. Also no AF-markdown PHB can be provisioned on this class during this interim.

Finally a new class is provisioned for Streaming Video. Testing has shown that there is a negligible difference in enabling WRED on this UDP-based traffic class, so although it remains an option, WRED is not enabled in these design examples.

The Cisco 7200 11-Class QoS Model is shown in Figure A-2. The inner circle shows how this model is backwards-compatible and consistent with the previous 8-Class model.

Figure A-2 Cisco 7200 11-Class QoS Model Example



The following configuration example shows the corresponding configuration for this Cisco 7200 11-Class Model. Keep in mind that this model is intended for links of speeds greater than 3 Mbps.

```
!
class-map match-all VOICE
 match ip dscp ef                ! QoS Baseline marking for Voice
class-map match-all INTERACTIVE-VIDEO
 match ip dscp af41 af42 af43         ! QoS Baseline marking Interactive-Video
class-map match-all IP-ROUTING
 match ip dscp cs6               ! QoS Baseline marking for IP Routing
class-map match-all NET-MGMT
 match ip dscp cs2               ! QoS Baseline marking for Network Mgmt
class-map match-all CALL-SIGNALING
 match ip dscp cs3               ! QoS Baseline marking for Call-Signaling
class-map match-all MISSION-CRITICAL-DATA
 match ip dscp af31 af32 af33         ! QoS Baseline marking Mission-Critical
class-map match-all TRANSACTIONAL-DATA
 match ip dscp af21 af22 af23         ! QoS Baseline marking Transactional-Data
class-map match-all BULK-DATA
 match ip dscp af11 af12 af13         ! QoS Baseline marking for Bulk-Data
```

```
class-map match-all STREAMING-VIDEO
 match ip dscp cs4                ! QoS Baseline marking Streaming-Video
class-map match-all SCAVENGER
 match ip dscp cs1                ! QoS Baseline marking for Scavenger
!
policy-map WAN-EDGE
 class VOICE
  priority percent 18                ! Voice gets LLQ - "dual-LLQ" design
 class INTERACTIVE-VIDEO
  priority percent 15                ! Int-Video gets LLQ - "dual-LLQ" design
 class IP-ROUTING
  bandwidth percent 3                ! Routing gets CBWFQ
 class NET-MGMT
  bandwidth percent 2                ! Network Management gets CBWFQ
 class CALL-SIGNALING
  bandwidth percent 5                ! Call-Signaling gets CBWFQ
 class MISSION-CRITICAL-DATA
  bandwidth percent 10               ! Critical Data gets CBWFQ
  random-detect dscp-based           ! Critical Data also gets DSCP-based WRED
 class TRANSACTIONAL-DATA
  bandwidth percent 7                ! Critical Data gets CBWFQ
  random-detect dscp-based           ! Critical Data also gets DSCP-based WRED
 class BULK-DATA
  bandwidth percent 4                ! Bulk Data gets CBWFQ
  random-detect dscp-based           ! Bulk Data also gets DSCP-based WRED
 class STREAMING-VIDEO
  bandwidth percent 10               ! Streaming-video gets CBWFQ
 class SCAVENGER
  bandwidth percent 1                ! Scavenger gets minimum CBWFQ
 class class-default
  bandwidth percent 25               ! Best Effort is protected with CBWFQ
  random-detect                   ! Best Effort also gets WRED (RED)
!
```

Cisco 7304 QoS Design

Cisco 7304 NSE-100 implements high-performance IP forwarding with services in the PXF processor on the NSE-100 forwarding engine. The following features are supported in the PXF path:

- Classification
- Marking
- Policing
- WRED
- LLQ
- CBWFQ
- Traffic Shaping

Currently, unlike regular IOS, NSE-100 PXF executes all MQC actions in a fixed order, regardless of the configured sequence. The general order of execution of QoS features is:

1. Input classification.
2. Only necessary if an input service-policy is present.
3. Input marking (**set**).
4. Input policing (**police**).

5. Output classification—The output classification is optimized in a way that it occurs if the packet was not already classified during input **or** if the packet header was modified during input processing (through marking/policing or some other feature, like NAT).
6. Output marking (**set**).
7. Output policing (**police**).
8. WRED (**random-detect**).
9. CBWFQ/LLQ/Traffic Shaping (bandwidth, priority, and shape).

It supports up to eight (8) different output queues per physical or logical interface. From these queues, two have special purposes:

- Crucial traffic queue—Dedicated for some types of internally (RP) generated vital traffic (mostly Layer 2 keep-alives).
- Default queue—Dedicated to traffic that does not match any user defined classes.

The other six (6) queues are the "user-defined queues," available for classes requiring queue related actions in the output service-policy.

The crucial traffic queue is a NSE-100-specific solution for handling those special types of internally-generated vital traffic that cannot be dropped. This queue has its own fixed parameters that cannot be configured by the user.

When no output service-policy is attached to an interface, only the crucial traffic and default queues are used. Those types of special locally-generated vital traffic go through the crucial traffic queue (usually very light traffic) and all other traffic goes through the default queue.

Classification

Classification on the Cisco 7304 NSE-100 can be applied based on:

- Layer 3 Criteria (IP packets only):
  - Access Control Lists (Turbo ACLs)
  - IP Precedence
  - IP DSCP
  - IP RTP port number/range
- MPLS Related Criteria (MPLS packets only):
  - MPLS experimental bits
- Internal settable variables (all packets):
  - QoS-Groups

Note When combining qos-group with other matching criteria within the same class-map, only the qos-group statements are considered; the other match statements are completely ignored.

Policing

On the NSE-100, the policing implementation is a single rate, 3-color policer. The traffic policer accepts the following parameters:

- **sustained rate** (bps)
- **normal burst** (bytes)
- **max burst** (bytes)

The term color refers to the following actions:

- **conform-action**—Traffic is less than the specified sustained rate
- **exceed-action**—Traffic exceeds the normal burst
- **violate-action**—Traffic exceeds the maximum burst

The result of these actions can be set to:

- **transmit**—Transmit the packet
- **drop**—Drop the packet

Or to remark the packets:

- **set-prec-transmit**—Rewrite the packet precedence and send it.
- **set-dscp-transmit**—Set the DSCP bits and send packet.
- **set-clp-transmit**—Set the ATM CLP bit and send packet.
- **set-frde-transmit**—Set the FR DE bit and send packet.
- **set-mpls-exp-imposition-transmit**—Set the MPLS experimental bit at imposition and send the packet.
- **set-qos-transmit**—Set the QoS group within the router and send packet.

Weighted Random Eary Detection (WRED)

WRED is enabled for congestion avoidance in a class through the **random-detect** command. Either the **bandwidth** or **shape** command must be already present in the class for IOS to allow the configuration of **random-detect**. With WRED configured, the queue size for that particular class is set to be the highest WRED maximum threshold x 2, rounded up to the next power of 2. The WRED maximum threshold ranges from 1 to 4096 packets and hence the queue size can be up to 8192 packets. By default, the max-thresholds for all drop profiles are set to 40, which gives a default queue size (rounding up to nearest power of 2) of 128 packets. The implementation on Cisco 7304 NSE-100 supports:

- IP precedence based WRED
- IP DSCP based WRED
- MPLS exp based WRED

This can give packets with low IP precedence, DSCP, or MPLS exp value a higher probability of being dropped than packets with high value. Up to 64 drop profiles are supported.

Note Discard-class based WRED is not supported.

Class-based Weighted Fair Queuing (CBWFQ)

CBWFQ is a congestion management implementation of WFQ. It provides support for configurable queuing for different traffic classes. A FIFO queue is allocated for each class containing queuing actions and traffic belonging to a class is directed to its proper queue. The minimum guaranteed bandwidth can be assigned in three different forms:

- A committed information rate in Kbps (**bandwidth** <kbps>)
- A percentage of the underlying link rate (**bandwidth percent** <percent>)
- A percentage of the bandwidth not allocated by the other classes (**bandwidth remaining percent** <percent>)

On the NSE-100, CBWFQ is implemented through the scheduler. While each form of the bandwidth command provides a means to allocate bandwidth to a traffic class, it is also the case that if a class does not use its allocated bandwidth (i.e., no traffic is offered to the class), then this excess bandwidth is available to other classes, which are then allowed to use this bandwidth and exceed their minimum allocation. On the NSE-100, by default, classes share excess bandwidth proportionally to the allocated bandwidth.

On the NSE-100, even though the CLI granularity is 1 Kbps, the actual bandwidth granularity is a factor of the link speed. More precisely, it is 1/65536 (1/64K) of the link speed. The value configured at CLI is internally rounded down to the closest multiple of 1/65536th of the link speed.

Examples:

- FastEthernet Link speed—100 Mbps => Granularity 100 Mbps / 65536 = ~ 1.53 Kbps
- GigEthernet Link speed—1 Gbps => Granularity 1 Gbps / 65536 = ~ 15.3 Kbps

Note The crucial traffic queue bandwidth is allocated after the bandwidth for the class queues. If, after the class queues bandwidths are allocated, there is still enough bandwidth left for the crucial traffic queue, it is simply allocated. On the other hand, if after allocating bandwidths for the class queues, there is not enough bandwidth left for the crucial traffic queue, all class queues bandwidths are internally and proportionally adjusted to the link speed minus the crucial traffic queue bandwidth.

Hierarchical Policies

A traffic policy is called hierarchical policy when it is defined using two or more policy-maps nested through the policy-map class sub-mode service-policy command. The operation of a hierarchical policy is recursive. When a hierarchical policy is used, the traffic matching a 1st-level policy-map class that has a child policy-map is subject to both the actions in this 1st-level class and the actions on the matching class on the child policy-map.

NSE-100 supports four different generic “flavors” of hierarchical traffic policy configurations, namely:

- Hierarchical traffic shaping for sub-interfaces
- Ingress hierarchical policing
- Queuing on parent, selective marking/policing on child
- Shaping on parent, selective marking/policing on child

The following configuration provides examples for both “flat” (core facing) as well as “hierarchical” (CE facing) policies on a 7304 NSE-100 PE:

```
class-map match-any realtime
```

```
  match mpls experimental topmost 5
class-map match-any realtime-2ce
  match ip precedence 5
class-map match-any bulk-data
  match mpls experimental topmost 1
class-map match-any bulk-data-2ce
  match ip precedence 1
class-map match-any bus-critical
  match mpls experimental topmost 3
class-map match-any trans-data
  match mpls experimental topmost 2
class-map match-any bus-critical-2ce
  match ip precedence 3
class-map match-any trans-data-2ce
  match ip precedence 2
class-map match-any control
  match mpls experimental topmost 6
  match mpls experimental topmost 7
class-map match-any video-2ce
  match ip precedence 4
class-map match-any video
  match mpls experimental topmost 4
class-map match-any control-2ce
  match ip precedence 7
  match ip precedence 6
!
policy-map q-core-out
  class realtime
    priority
   police cir 300000000
     conform-action transmit
     exceed-action drop
  class control
    bandwidth remaining percent 14
    random-detect
    random-detect precedence 6 300 1500 1
    random-detect precedence 7 300 1500 1
  class bus-critical
    bandwidth remaining percent 14
    random-detect
    random-detect precedence 3 300 1500 1
  class trans-data
    bandwidth remaining percent 14
    random-detect
    random-detect precedence 2 300 1500 1
  class video
    bandwidth remaining percent 14
    random-detect
    random-detect precedence 4 300 1500 1
  class bulk-data
    bandwidth remaining percent 7
    random-detect
    random-detect precedence 1 300 1500 1
  class class-default
    bandwidth remaining percent 36
    random-detect
    random-detect precedence 0 300 1500 1
policy-map q-2ce-out-1
  class control-2ce
    bandwidth percent 14
    random-detect
    random-detect precedence 6 300 1500 1
    random-detect precedence 7 300 1500 1
  class bus-critical-2ce
```

```
    bandwidth percent 14
    random-detect
    random-detect precedence 3 300 1500 1
  class trans-data-2ce
    bandwidth percent 14
    random-detect
    random-detect precedence 2 300 1500 1
  class video-2ce
    bandwidth percent 14
    random-detect
    random-detect precedence 4 300 1500 1
  class bulk-data-2ce
    bandwidth percent 7
    random-detect
    random-detect precedence 1 300 1500 1
  class class-default
    bandwidth percent 36
    random-detect
    random-detect precedence 0 300 1500 1
policy-map q-2ce-out-data
  class class-default
    shape average 650000000
   service-policy q-2ce-out-1
policy-map q-2ce-out-2
  class realtime-2ce
    priority
   police cir 300000000
     conform-action transmit
     exceed-action drop
  class control-2ce
    random-detect
    random-detect precedence 6 300 1500 1
    bandwidth percent 3
  class class-default
    random-detect
    bandwidth percent 2
policy-map q-2ce-out-voice
  class class-default
    shape average 350000000
   service-policy q-2ce-out-2
!
interface GigabitEthernet3/0/1
 description Core Facing
 ip address 125.1.103.22 255.255.255.252
 mpls ip
 service-policy output q-core-out
!
interface GigabitEthernet3/1/0.1
 description BLUE-DATA - To CE
 encapsulation dot1Q 175
 ip vrf forwarding blue-data
 ip address 125.1.103.49 255.255.255.252
 service-policy output q-2ce-out-data
!
interface GigabitEthernet3/1/0.2
 description BLUE-VOICE - To CE
 encapsulation dot1Q 176
 ip vrf forwarding blue-voice
 ip address 125.1.103.53 255.255.255.252
 service-policy output q-2ce-out-voice
```

Cisco 7600 QoS Design

The Cisco 7600 series routers perform QoS in hardware. Classification, marking, and policing are performed within the Policy Feature Card (PFC3B or PFC3BXL, hereafter referred to as PFC3B)) and queuing and dropping (tail-drop or WRED) is performed within line card hardware. As such there is no incremental CPU load when enabling such features, even at GE and 10GE speeds.

The following sections examine the configuration of Uniform Mode MPLS DiffServ Tunneling on the Cisco 7600, which is performed within the PFC3B, the trust states of various MPLS label pushing, swapping, and popping operations, and finally line card-specific queuing designs.

Cisco 7600—Uniform Mode MPLS DiffServ Tunneling

Configuring Uniform Mode MPLS DiffServ Tunneling on the Cisco 7600 consists of two main parts: trusting IPP or DSCP (recommended) on the PE-to-CE interface (applied in the ingress direction from the CE) and applying an **mpls propagate-cos command** to overwrite the IPP value with the last MPLS EXP value (applied in the egress direction towards the CE).

The following is an example of the Cisco 7600 Uniform Mode configuration.

```
C7600(config)# interface GE-WAN 3/1
C7600(config-if)# mls qos trust dscp
C7600(config-if)# interface GE-WAN 3/2.32
C7600(config-if)# mpls propagate-cos
```

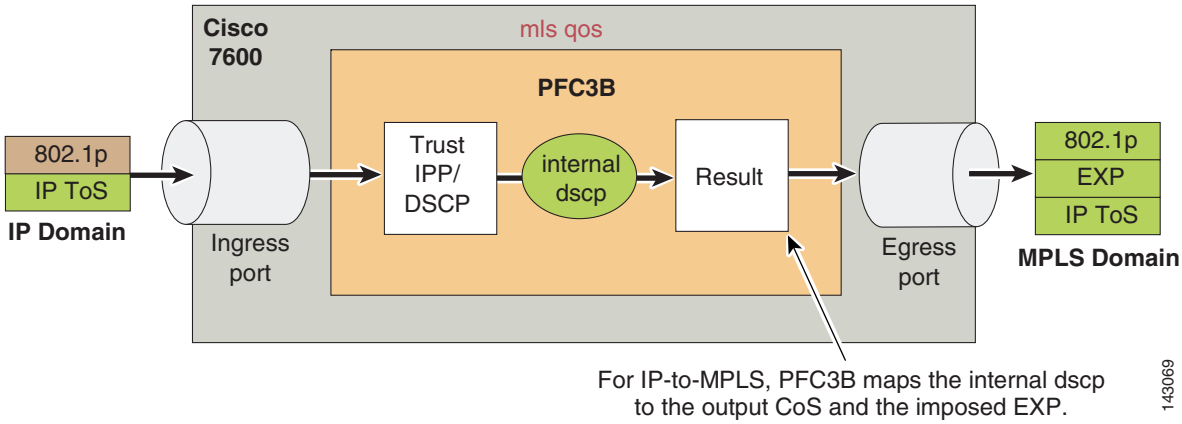
Cisco 7600—Trust States and Internal DSCP Generation

The Cisco 7600 performs queuing and dropping decisions based on the concept of an “internal DSCP.” The Cisco 7600 generates an internal DSCP for all packets/frames, regardless of whether they are IP or otherwise. For example, this internal DSCP may be derived by setting a port state to **trust dscp**, in which case the internal DSCP is set to match the packet DSCP. Alternatively, in an 802.1Q/p environment, the port can be set to **trust cos**, in which case the internal DSCP is generated by accepting the CoS marking value and performing a conversion to DSCP by means of the CoS-to-DSCP mapping table.

In MPLS environments the following rules apply to trust and the generation of the internal DSCP (used for queuing and dropping):

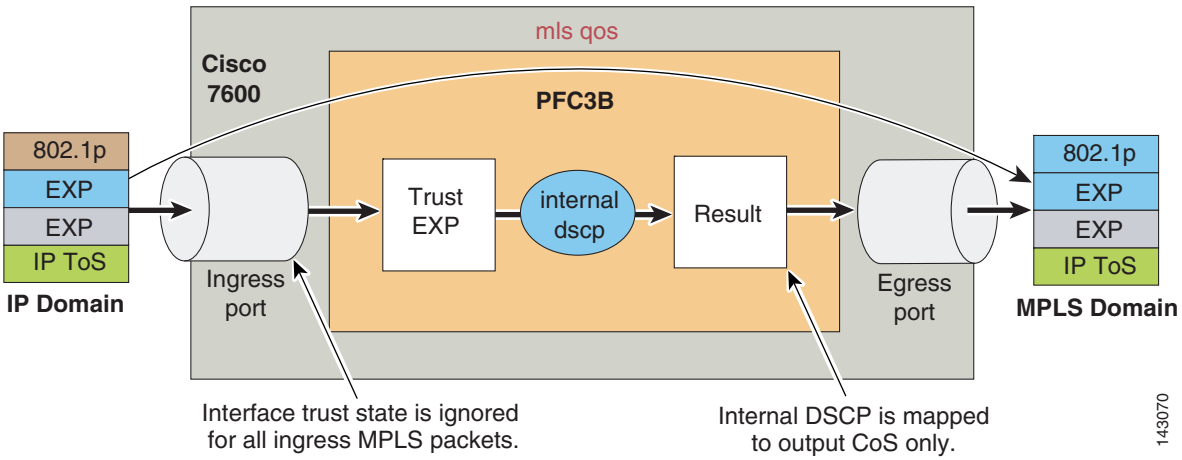
- When PFC3B receives an IP packet (IP-to-IP or IP-to-MPLS), it uses the input interface trust state and, if configured, the **policy-map trust** command. During MPLS label imposition, for packets received on an interface with trust IPP or trust dscp, PFC3B maps the IPP/DSCP to the internal DSCP. It then maps the internal DSCP to the imposed EXP and the output CoS. It always preserves the underlying IP ToS as shown in Figure A-3.

Figure A-3 Cisco 7600 MPLS Label Imposition (Pushing) Trust



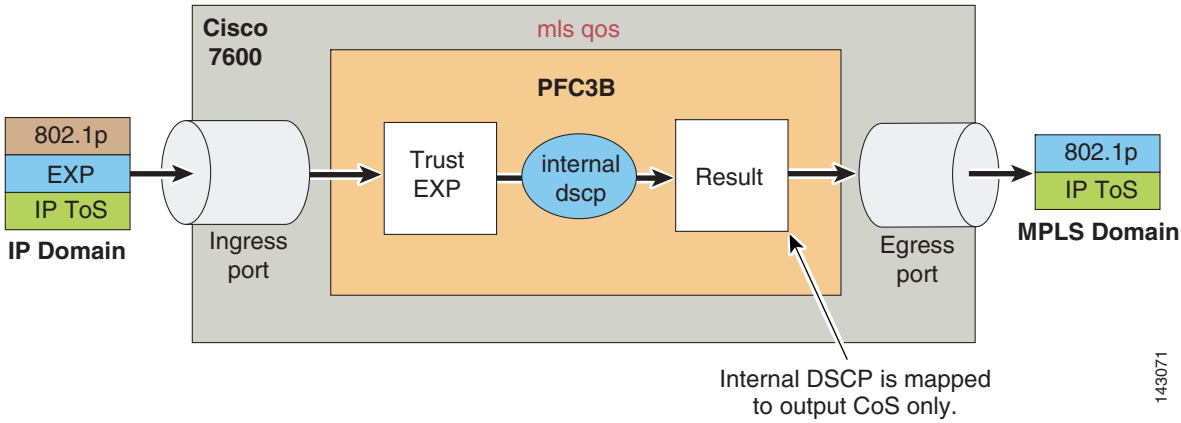
- When PFC3B receives an MPLS packet to be swapped (MPLS-to-MPLS), it trusts EXP; the interface trust state and the **policy-map trust** command have no effect. During swapping, PFC3B trusts the topmost EXP and maps it to the internal DSCP. During the swap, it copies EXP from the swapped-off label to the swapped-on label. After the swap, it maps the internal DSCP to the egress frame CoS as shown in Figure A-4.

Figure A-4 Cisco 7600 MPLS Label Switching (Swapping) Trust



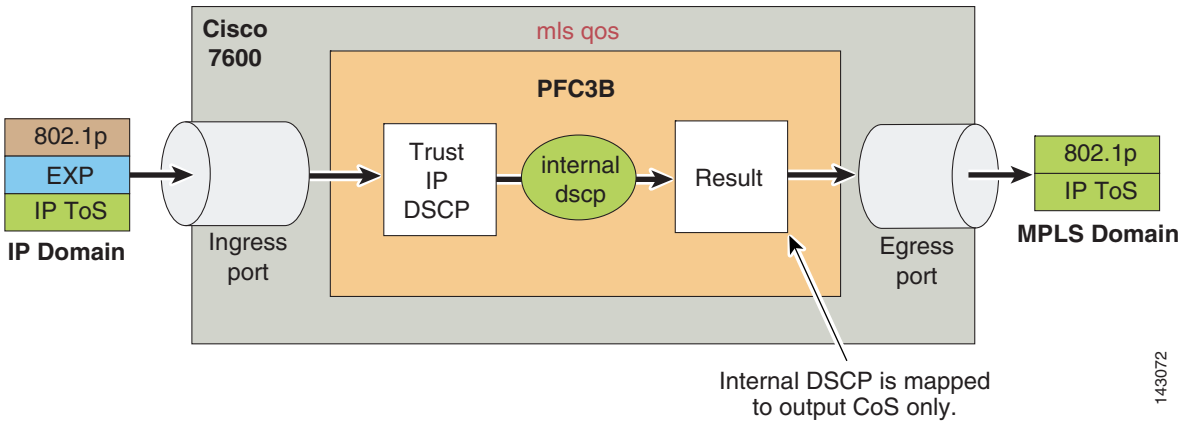
- When PFC3B receives an MPLS packet to be popped (MPLS-to-IP), trust depends on the type of label; however, in all cases, the interface trust state and the **policy-map trust** command have no effect.
- Non-aggregate label—PFC3B/PFC3BXL trusts EXP in the topmost label. PFC3B trusts EXP and maps it to the internal DSCP. By default, PFC3B discards the popped EXP and does not propagate it to the exposed IP ToS. After the pop, it maps the internal DSCP to the egress frame CoS as shown in Figure A-5.

Figure A-5 Cisco 7600 MPLS Label Disposition (Popping) Trust—Case 1 (Non-Aggregate Label)



- Aggregate label in VPN CAM—PFC3B/PFC3BXL trusts IP DSCP.
- Aggregate label not in VPN CAM—PFC3B/PFC3BXL trusts IP DSCP (after recirculation). PFC3B trusts the exposed IP ToS and maps it to the internal DSCP. By default, PFC3B discards the popped EXP and does not propagate it to the exposed IP ToS. After the pop, it maps the internal DSCP to the egress frame CoS as shown in Figure A-6.

Figure A-6 Cisco 7600 MPLS Label Disposition (Popping) Trust—Case 1 (Non-Aggregate Label)



For example, if an IP packet is received and the interface is set to trust dscp (recommended), then the internal DSCP is set to match the packet DSCP. When queued on egress, even though a CoS-to-queue mapping is used, this actually represents an internal DSCP-to-queue mapping, such that mapping CoS 2-to-queue 2 actually represents assigning any packets that have an internal DSCP value of 16 through 23 to queue 2. Continuing the example, if a MPLS packet is received and is being forwarded as an MPLS packet (swapping function of a P Cisco 7600 router), then the EXP value is trusted, the internal DSCP value is calculated from the EXP-to-DSCP mapping table, and the packet is queued based on the internal DSCP value (via the CoS-to-queue mapping table). Concluding the example, if a non-aggregate MPLS packet is received and is to be forwarded as an IP packet (popping function of a PE Cisco 7600 router), then the internal DSCP value is calculated by trusting the topmost label EXP value and applying the EXP-to-DSCP mapping, and finally the packet is queued based on the internal DSCP value (via the CoS-to-queue mapping table).

Cisco 7600—Queuing Design

Cisco 7600 line cards support both ingress and egress queuing; furthermore, these ingress and egress queuing structures vary by line card. Ingress congestion implies that the combined ingress rates of traffic exceed the switch fabric channel speed, and thus need to be queued simply to gain access to the switching fabric. On newer platforms, such as the Cisco 7600 Sup720, this means that a combined ingress rate of up to 40 Gbps per slot is required to create such an event. However to obviate such an extreme event, the Catalyst 7600 schedules ingress traffic through the receive queues based on CoS values. In the default configuration, the scheduler assigns all traffic with CoS 5 to the strict priority queue (if present); in the absence of a strict priority queue, the scheduler assigns all traffic to the standard queues. All other traffic is assigned to the standard queues (with higher CoS values being assigned preference over lower CoS values, wherever supported). Thus even in the highly unlikely event of ingress congestion, the default settings for the receive queues of the Cisco 7600 line cards are more than adequate to protect VoIP and network control traffic. Therefore the focus of this section is on Cisco 7600 egress/transmit queuing design recommendations.

The following five egress queuing structures are currently supported on Cisco 7600 Gigabit, Ten-Gigabit, or 10/100/1000 line cards:

- 1P2Q1T—Indicates one strict priority queue and two standard queues, each with one configurable WRED-drop threshold and one non-configurable (100 percent) tail-drop threshold.
- 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 or tail-drop threshold and one non-configurable (100 percent) tail-threshold.
- 1P3Q8T—Indicates one strict priority queue and three standard queues, each with eight configurable WRED-drop or tail-drop thresholds.
- 1P7Q8T—Indicates one strict priority queue and seven standard queues, each with eight configurable WRED-drop or tail-drop thresholds.

Table A-1 and Table A-2 summarize these queuing structures by line card.

T

Table A-1 Cisco 7600 Classic and CEF256 Line Cards and Queuing Structures

Classic/ CEF256 Ethernet Modules	Description	Rx Queuing	Tx Queuing	Buffer Size
WS-X6148-GE-TX	48-Port 10/100/1000 RJ-45 Module	1Q2T	1P2Q2T	1MB per 8 ports
WS-X6148V-GE-TX	48-Port 10/100/1000 Inline Power RJ-45 Module	1Q2T	1P2Q2T	1MB per 8 ports
WS-X6316-GE-TX	16-Port 1000TX GigabitEthernet RJ-45 Module	1P1Q4T	1P2Q2T	512KB per port
WS-X6408A-GBIC	8-Port GigabitEthernet Module (with enhanced QoS; requires GBICs)	1P1Q4T	1P2Q2T	512KB per port
WS-X6416-GBIC	16-Port GigabitEthernet Module (requires GBICs)	1P1Q4T	1P2Q2T	512KB per port
WS-X6416-GE-MT	16-Port GigabitEthernet MT-RJ Module	1P1Q4T	1P2Q2T	512KB per port
WS-X6501-10GEX4	10 GigabitEthernet Module	1P1Q8T	1P2Q1T	64MB per port
WS-X6502-10GE	10 GigabitEthernet Base Module (requires OIM)	1P1Q8T	1P2Q1T	64MB per port

Table A-1 Cisco 7600 Classic and CEF256 Line Cards and Queuing Structures (continued)

WS-X6516A-GBIC	GigabitEthernet Module (fabric-enabled; requires GBICs)	1P1Q4T	1P2Q2T	1MB per port
WS-X6516-GBIC	GigabitEthernet Module (fabric-enabled; requires GBICs)	1P1Q4T	1P2Q2T	512KB per port
WS-X6516-GE-TX	16-Port GigabitEthernet Copper Module; (crossbar-enabled)	1P1Q4T	1P2Q2T	512KB per port
WS-X6524-100FX-MM	24-Port 100FX MT-RJ Module (Fabric-Enabled)	1P1Q0T	1P3Q1T	1MB per port
WS-X6548-RJ-21	48-Port 10/100 RJ-21 Module (fabric- enabled)	1P1Q0T	1P3Q1T	1MB per port
WS-X6548-RJ-45	48-Port 10/100 RJ-45 Module (crossbar-enabled)	1P1Q0T	1P3Q1T	1MB per port
WS-X6548V-GE-TX	48-Port 10/100/1000 Inline Power RJ- 45 Module (fabric-enabled)	1Q2T	1P2Q2T	1MB per 8 ports
WS-X6548-GE-TX	48-Port 10/100/1000 RJ-45 Module (fabric-enabled)	1Q2T	1P2Q2T	1MB per 8 ports
WS-X6816-GBIC	16-Port GigabitEthernet Module (fabric-enabled; requires GBICs)	1P1Q4T	1P2Q2T	512KB per port

Table A-2 Cisco 7600 CEF720 Line Cards and Queuing Structures

C2 (xCEF720) Modules	Description	Rx-Queuing	Tx-Queuing	Buffer Size
WS-X6704-10GE	4-Port 10 GigabitEthernet Module	1Q8T (8Q8T with DFC3a)	1P7Q8T	16MB per port
WS-X6724-SFP	24-Port GigabitEthernet SFP Module	1Q8T (2Q8T with DFC3a)	1P3Q8T	1MB per port
WS-X6748-GE-TX	48-Port 10/100/1000 RJ-45 Module	1Q8T (2Q8T with DFC3a)	1P3Q8T	1MB per port
WS-X6748-SFP	48-Port GigabitEthernet SFP Module1	1Q8T (2Q8T with DFC3a)	1P3Q8T	1MB per port

Note For any newer line cards not on this list, the queuing structure can be ascertained by the **show queuing interface verification** command.

Cisco 7600 1P2Q1T 10GE Queuing Design

Under the 1P2Q1T queuing model, buffer space can be allocated as follows: 30 percent for Scavenger/Bulk plus Best Effort queue (Q1) and 40 percent for Q2, the Critical Data queue (assigning buffer space for Q3, the PQ in this model, is not supported on this line card).

The WRR weights for Q1 and Q2 (for dividing the remaining bandwidth, after the priority queue has been fully serviced) can be set to 30:70 respectively for Q1:Q2.

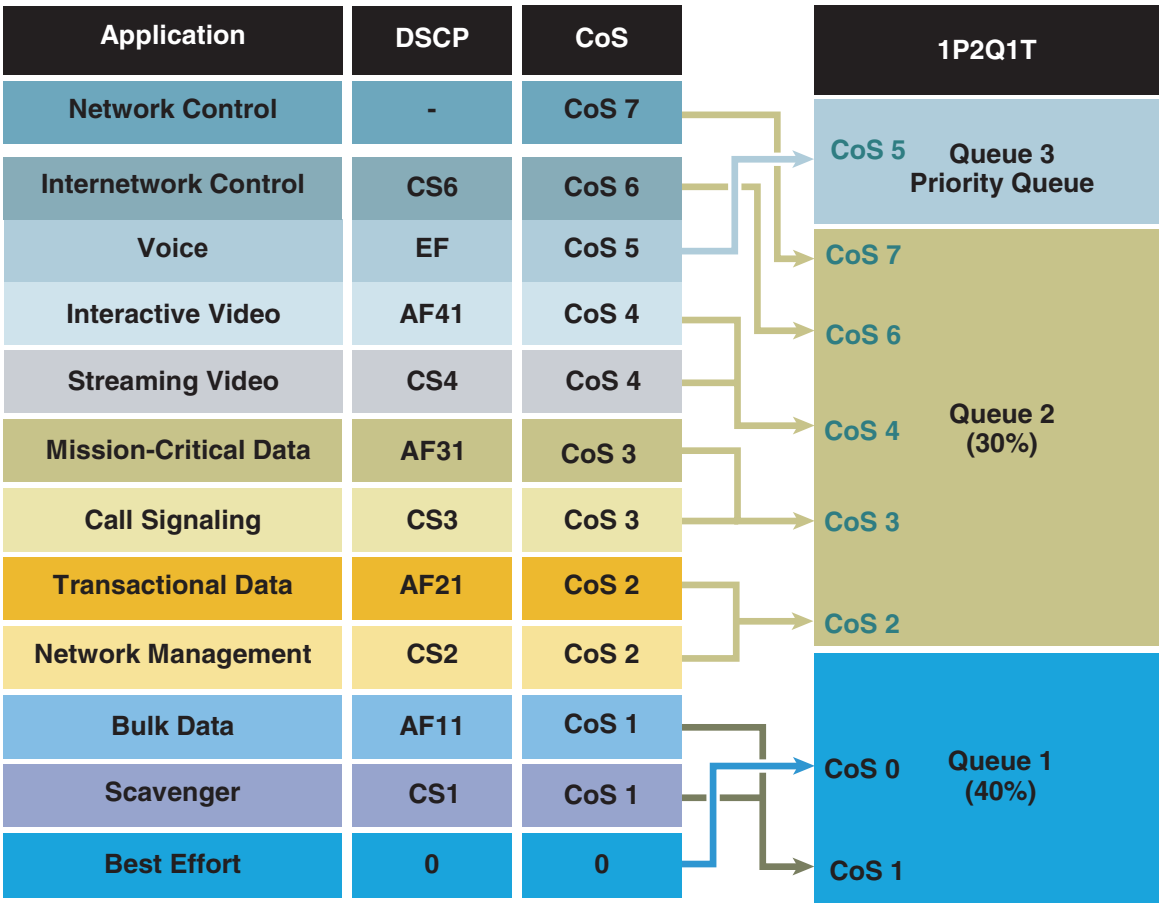
The Q1T1 WRED threshold can be set to 80:100 and the Q2T1 WRED threshold can be set to 80:100.

After these WRED thresholds have been altered, the following assignments can be made:

- CoS 1 (Scavenger/Bulk) and CoS 0 (Best Effort) to Q1T1.
- CoS 2 (Network Management and Transactional Data), CoS 3 (Call-Signaling and Mission-Critical Data), CoS 4 (Interactive and Streaming Video) ,and CoS 6 and 7 (Internetwork and Network Control) to Q2T1.
- CoS 5 (VoIP) to Q3 (the PQ).

These Cisco 7600 1P2Q1T queuing recommendations are illustrated in Figure A-7.

Figure A-7 Cisco 7600 1P2Q1T Queuing Model



The Cisco 7600 commands to configure 1P2Q1T queuing recommendations are shown in the following configuration example.

```
C7600(config)#interface TenGigabitEthernet1/1
C7600(config-if)# wrr-queue queue-limit 30 40
! Sets the buffer allocations to 30% for Q1 and 40% for Q2
C7600(config-if)# wrr-queue bandwidth 30 70
! Sets the WRR weights for 30:70 (Q1:Q2) bandwidth servicing
C7600(config-if)#
C7600(config-if)# wrr-queue random-detect min-threshold 1 80
! Sets Min WRED Threshold for Q1T1 to 80%
C7600(config-if)# wrr-queue random-detect max-threshold 1 100
! Sets Max WRED Threshold for Q1T1 to 100%
C7600(config-if)# wrr-queue random-detect min-threshold 2 80
! Sets Min WRED Threshold for Q2T1 to 80%
C7600(config-if)# wrr-queue random-detect max-threshold 2 100
! Sets Max WRED Threshold for Q2T1 to 100%
C7600(config-if)#
C7600(config-if)# wrr-queue cos-map 1 1 1 0
! Assigns Scavenger/Bulk and Best Effort to Q1 WRED Threshold 1
C7600(config-if)# wrr-queue cos-map 2 1 2 3 4 6 7
! Assigns CoS 2,3,4,6 and 7 to Q2 WRED Threshold 1
C7600(config-if)# priority-queue cos-map 1 5
! Assigns VoIP to PQ (Q3)
C7600(config-if)#end
C7600(config-if)#
```

Cisco 7600 1P2Q2T GE Queuing Design

On the Cisco 7600, setting the size of the priority queue is not supported on any queuing structure with one exception: the 1P2Q2T structure, where the priority queue (Q3) is indirectly set to equal the Q2 size.

Under a 1P2Q2T model, buffer space can be allocated as follows: 40 percent for Q1 (the Scavenger/Bulk plus Best Effort queue) and 30 percent for Q2 (the Critical Data queue); therefore Q3 (the priority queue) is also indirectly set to 30 percent (to equal the size of Q2).

The WRR weights for Q1 and Q2 (for dividing the remaining bandwidth, after the priority queue has been fully serviced) remain at 30:70 respectively for Q1:Q2.

Under the 1P2Q2T model, each WRED threshold is defined with a lower and upper limit. Therefore the first WRED threshold for Q1 can be set to 40:80, so that Scavenger/Bulk Data traffic can be WRED-dropped if Q1 hits 40 percent and can be tail-dropped if Q1 exceeds 80 percent of its capacity; this prevents Scavenger/Bulk Data from drowning out Best Effort traffic in Q1. The second WRED threshold for Q1 can be set to 80:100 to provide congestion avoidance for Best Effort traffic.

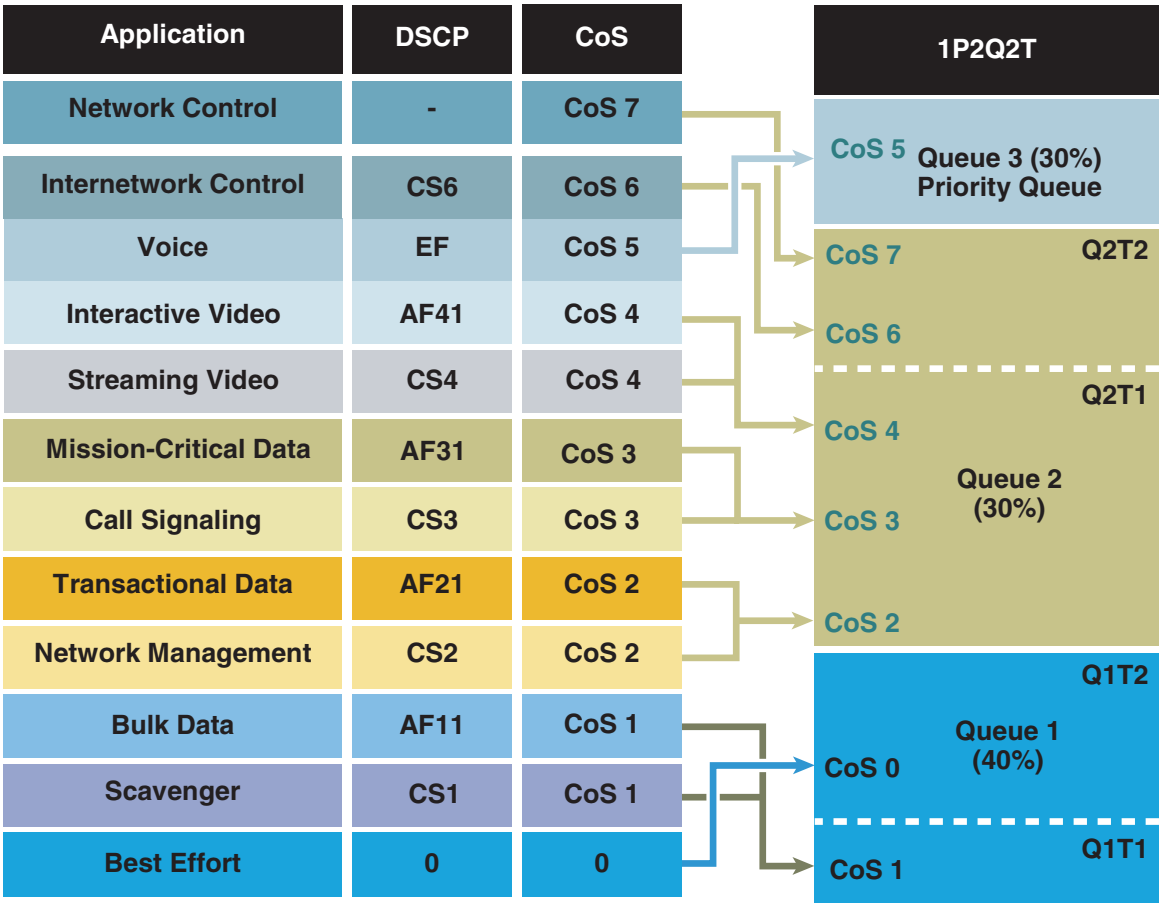
Similarly, the first WRED threshold of Q2 can be set to 70:80 and the second can be set to 80:100. In this manner, congestion avoidance is provided for all traffic types in Q2 and there is always room in the queue to service Network and Internetwork Control traffic.

After the queues have been defined as above, the following assignments can be made:

- CoS 1 (Scavenger/Bulk) to Q1T1
- CoS 0 (Best Effort) to Q1T2
- CoS 2 (Network Management and Transactional Data), CoS 3 (Call-Signaling and Mission-Critical Data), and CoS 4 (Interactive and Streaming Video) to Q2T1
- CoS 6 and 7 (Internetwork and Network Control) to Q2T2
- CoS 5 (VoIP) to Q3T1 (the PQ)

These 1P2Q2T queuing recommendations are illustrated in Figure A-8.

Figure A-8 Cisco 7600 1P2Q2T Queuing Model



The Cisco 7600 commands to configure 1P2Q2T queuing recommendations are shown in the following configuration example.

```
C7600(config)#interface range GigabitEthernet4/1 - 8
C7600(config-if-range)# wrr-queue queue-limit 40 30
! Sets the buffer allocations to 40% for Q1 and 30% for Q2
! Indirectly sets PQ (Q3) size to equal Q2 (which is set to 30%)
C7600(config-if-range)# wrr-queue bandwidth 30 70
! Sets the WRR weights for 30:70 (Q1:Q2) bandwidth servicing
C7600(config-if-range)#
C7600(config-if-range)# wrr-queue random-detect min-threshold 1 40 80
! Sets Min WRED Thresholds for Q1T1 and Q1T2 to 40 and 80
C7600(config-if-range)# wrr-queue random-detect max-threshold 1 80 100
! Sets Max WRED Thresholds for Q1T1 and Q1T2 to 80 and 100
C7600(config-if-range)#
C7600(config-if-range)# wrr-queue random-detect min-threshold 2 70 80
! Sets Min WRED Thresholds for Q2T1 and Q2T2 to 70 and 80
C7600(config-if-range)# wrr-queue random-detect max-threshold 2 80 100
! Sets Max WRED Thresholds for Q2T1 and Q2T2 to 80 and 100
C7600(config-if-range)#
C7600(config-if-range)# wrr-queue cos-map 1 1 1
! Assigns Scavenger/Bulk to Q1 WRED Threshold 1
C7600(config-if-range)# wrr-queue cos-map 1 2 0
! Assigns Best Effort to Q1 WRED Threshold 2
C7600(config-if-range)# wrr-queue cos-map 2 1 2 3 4
! Assigns CoS 2,3,4 to Q2 WRED Threshold 1
```

```
C7600(config-if-range)# wrr-queue cos-map 2 2 6 7
! Assigns Network/Internetwork Control to Q2 WRED Threshold 2
C7600(config-if-range)#
C7600(config-if-range)# priority-queue cos-map 1 5
! Assigns VoIP to PQ
C7600(config-if-range)#end
C7600#
```

Cisco 7600 1P3Q1T GE Queuing Design

Tuning the transmit size ratios is not supported in the Cisco 7600 1P3Q1T queuing structure. Furthermore under this queuing model Q4 becomes the priority queue.

The WRR weights for the standard queues (Q1, Q2, Q3) for dividing the remaining bandwidth, after the priority queue has been fully serviced, can be set to 5:25:70 respectively for Q1:Q2:Q3.

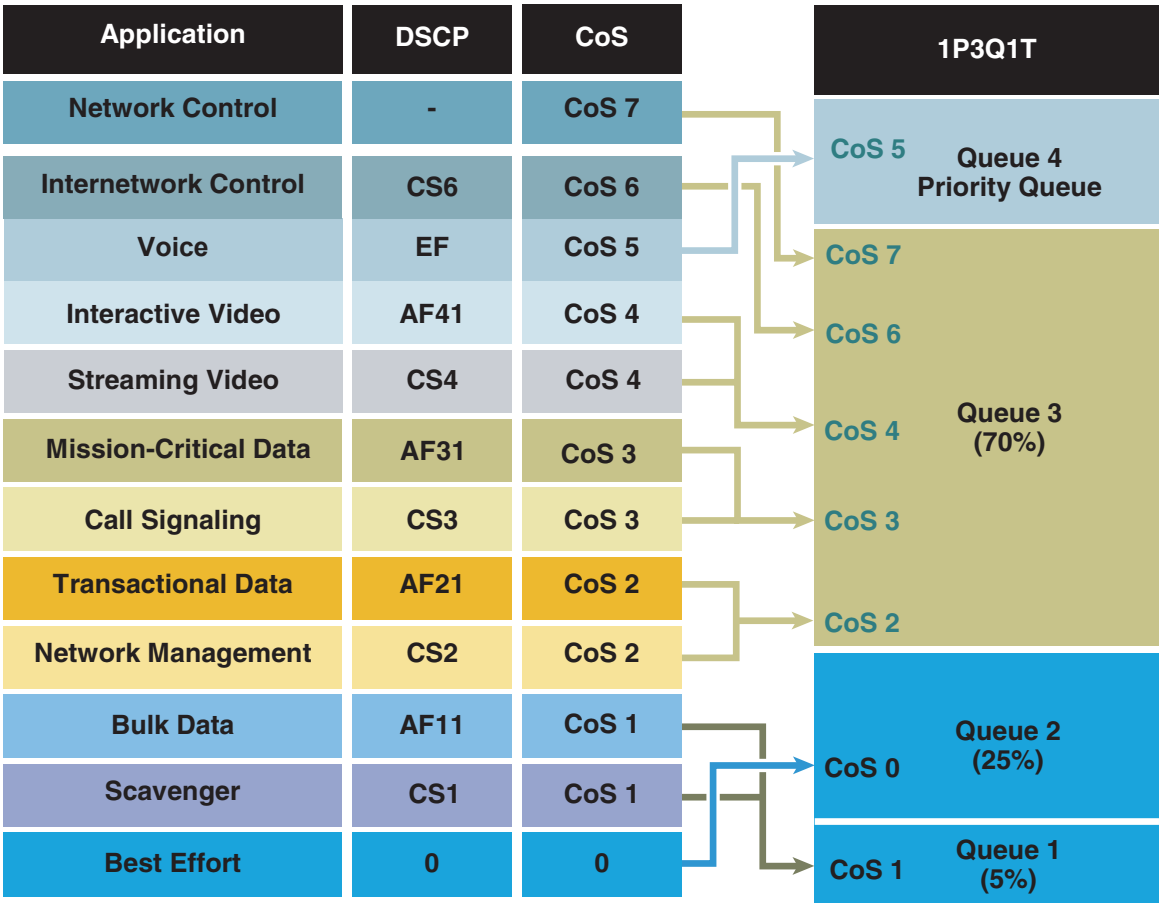
The Cisco 7600 1P3Q1T, 1P3Q8T, and 1P7Q8T queuing structures can be configured to use tail-drop or WRED. By default, WRED is disabled. Therefore, it is good practice to always explicitly enable WRED on a queue before setting WRED thresholds for these queuing structures. The WRED thresholds for all three preferential queues can be set to 80:100.

After the queues and thresholds have been defined as above, the following assignments can be made:

- CoS 1 (Scavenger/Bulk) to Q1T1
- CoS 0 (Best Effort) to Q2T1
- CoS 2 (Network Management and Transactional Data), CoS 3 (Call-Signaling and Mission-Critical Data), CoS 4 (Interactive and Streaming Video), and CoS 6 and 7 (Internetwork and Network Control) to Q3T1
- CoS 5 (VoIP) to Q4 (the PQ)

These 1P3Q1T queuing recommendations are illustrated in Figure A-9.

Figure A-9 Cisco 7600 1P3Q1T Queuing Model



The Cisco 7600 commands to configure 1P3Q1T queuing recommendations are shown in the following configuration example.

```
C7600(config)# interface range FastEthernet3/1 - 48
C7600(config-if)# wrr-queue bandwidth 5 25 70
! Sets the WRR weights for 5:25:70 (Q1:Q2:Q3) bandwidth servicing
C7600(config-if)#
C7600(config-if)#
C7600(config-if-range)# wrr-queue random-detect 1
! Enables WRED on Q1
C7600(config-if-range)# wrr-queue random-detect 2
! Enables WRED on Q2
C7600(config-if-range)# wrr-queue random-detect 3
! Enables WRED on Q3
C7600(config-if)#
C7600(config-if)# wrr-queue random-detect min-threshold 1 80
! Sets Min WRED Threshold for Q1T1 to 80%
C7600(config-if)# wrr-queue random-detect max-threshold 1 100
! Sets Max WRED Threshold for Q1T1 to 100%
C7600(config-if)#
C7600(config-if)# wrr-queue random-detect min-threshold 2 80
! Sets Min WRED Threshold for Q2T1 to 80%
C7600(config-if)# wrr-queue random-detect max-threshold 2 100
! Sets Max WRED Threshold for Q2T1 to 100%
C7600(config-if)#
C7600(config-if)# wrr-queue random-detect min-threshold 3 80
```

```
! Sets Min WRED Threshold for Q3T1 to 80%
C7600(config-if)# wrr-queue random-detect max-threshold 3 100
! Sets Max WRED Threshold for Q3T1 to 100%
C7600(config-if)#
C7600(config-if)# wrr-queue cos-map 1 1 1
! Assigns Scavenger/Bulk to Q1 WRED Threshold 1 (80:100)
C7600(config-if)# wrr-queue cos-map 2 1 0
! Assigns Best Effort to Q2 WRED Threshold 1 (80:100)
C7600(config-if)# wrr-queue cos-map 3 1 2 3 4 6 7
! Assigns CoS 2,3,4,6 and 7 to Q3 WRED Threshold 1 (80:100)
C7600(config-if)# priority-queue cos-map 1 5
! Assigns VoIP to PQ (Q4)
C7600(config-if)#end
C7600#
```

Cisco 7600 1P3Q8T GE Queuing Design

The Cisco 7600 1P3Q8T queuing structure is identical to the 1P3Q1T structure except that it has eight tunable WRED thresholds per queue (instead of one) and it also supports tuning the transmit size ratios.

Under a 1P3Q8T model buffer space can be allocated as follows: 5 percent for the Scavenger/Bulk queue (Q1), 25 percent for the Best Effort queue (Q2), 40 percent for the Critical Data queue (Q3), and 30 percent for the strict priority queue (Q4).

The WRR weights for the standard queues (Q1, Q2, Q3) for dividing the remaining bandwidth, after the priority queue has been fully serviced, can be set to 5:25:70 respectively for Q1:Q2:Q3.

The tunable WRED threshold for Q1 can be set to 80:100 to provide congestion avoidance to Scavenger/Bulk Data traffic. The WRED threshold for Q2 similarly can be set to 80:100 to provide congestion avoidance on all Best Effort flows.

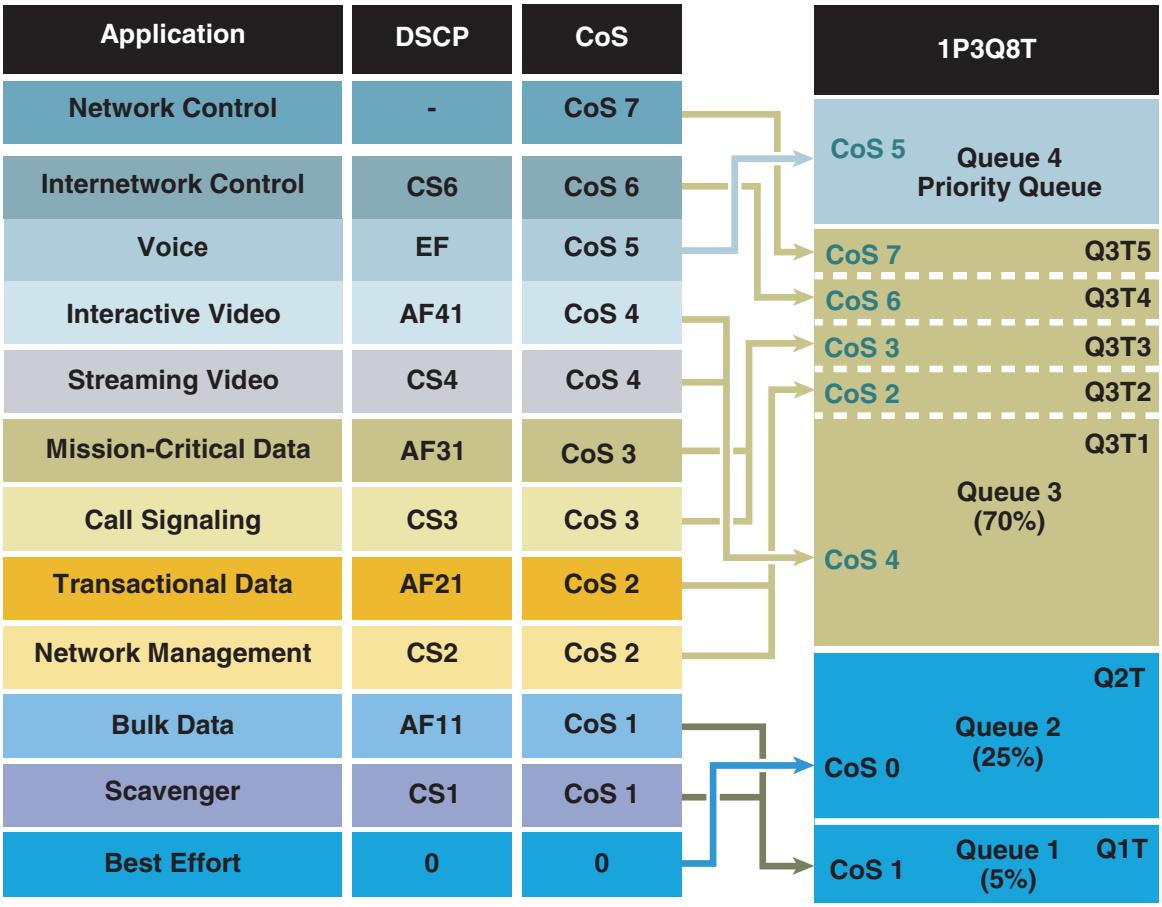
The 1P3Q8T queuing structure support for up to eight WRED thresholds per queue allows for additional QoS granularity for the applications sharing Q3. Because only five discrete CoS values are sharing this queue, only five of eight thresholds need to be defined for subqueue QoS. For example, Q3T1 can be set to 50:60, Q3T2 can be set to 60:70, Q3T3 can be set to 70:80, Q3T4 can be set to 80:90, and Q3T5 can be set to 90:100.

After the queues and thresholds have been defined as above, the following assignments can be made:

- CoS 1 (Scavenger/Bulk) to Q1T1
- CoS 0 (Best Effort) to Q2T1
- CoS 4 (Interactive and Streaming Video) to Q3T1
- CoS 2 (Network Management and Transactional Data) to Q3T2
- CoS 3 (Call-Signaling and Mission-Critical Data) to Q3T3
- CoS 6 (Internetwork Control) to Q3T4
- CoS 7 (Internetwork and Network Control) to Q3T5
- CoS 5 (VoIP) to Q4 (the PQ)

These Cisco 7600 1P3Q8T queuing recommendations are illustrated in Figure A-10.

Figure A-10 Cisco 7600 1P3Q8T Queuing Model



The Cisco 7600 commands to configure 1P3Q8T queuing recommendations are shown in the following configuration example.

```
C7600(config)# interface range GigabitEthernet1/1 - 48
C7600(config-if)# wrr-queue queue-limit 5 25 40
! Allocates 5% for Q1, 25% for Q2 and 40% for Q3
C7600(config-if)# wrr-queue bandwidth 5 25 70
! Sets the WRR weights for 5:25:70 (Q1:Q2:Q3) bandwidth servicing
C7600(config-if)#
C7600(config-if-range)# wrr-queue random-detect 1
! Enables WRED on Q1
C7600(config-if-range)# wrr-queue random-detect 2
! Enables WRED on Q2
C7600(config-if-range)# wrr-queue random-detect 3
! Enables WRED on Q3
C7600(config-if)#
C7600(config-if)# wrr-queue random-detect min-threshold 1 80
100 100 100 100 100 100 100
! Sets Min WRED Threshold for Q1T1 to 80% and all others to 100%
C7600(config-if)# wrr-queue random-detect max-threshold 1 100
100 100 100 100 100 100 100
! Sets Max WRED Threshold for Q1T1 to 100% and all others to 100%
C7600(config-if)#
C7600(config-if)# wrr-queue random-detect min-threshold 2 80
100 100 100 100 100 100 100
! Sets Min WRED Threshold for Q2T1 to 80% and all others to 100%
```

```
C7600(config-if)# wrr-queue random-detect max-threshold 2 100
100 100 100 100 100 100 100
! Sets Max WRED Threshold for Q2T1 to 100% and all others to 100%
C7600(config-if)#
C7600(config-if)# wrr-queue random-detect min-threshold 3 50
60 70 80 90 100 100 100
! Sets Min WRED Threshold for Q3T1 to 50%, Q3T2 to 60%,
! Q3T3 to 70%, Q3T4 to 80%, Q3T5 to 90% and all others to 100%
C7600(config-if)# wrr-queue random-detect max-threshold 3 60
70 80 90 100 100 100 100
! Sets Max WRED Threshold for Q3T1 to 60%, Q3T2 to 70%,
! Q3T3 to 80%, Q3T4 to 90%, Q3T5 to 100% and all others to 100%
C7600(config-if)#
C7600(config-if)# wrr-queue cos-map 1 1 1
! Assigns Scavenger/Bulk to Q1 WRED Threshold 1
C7600(config-if)# wrr-queue cos-map 2 1 0
! Assigns Best Effort to Q2 WRED Threshold 1
C7600(config-if)# wrr-queue cos-map 3 1 4
! Assigns Video to Q3 WRED Threshold 1
C7600(config-if)# wrr-queue cos-map 3 2 2
! Assigns Net-Mgmt and Transactional Data to Q3 WRED T2
C7600(config-if)# wrr-queue cos-map 3 3 3
! Assigns call signaling and Mission-Critical Data to Q3 WRED T3
C7600(config-if)# wrr-queue cos-map 3 4 6
! Assigns Internetwork-Control (IP Routing) to Q3 WRED T4
C7600(config-if)# wrr-queue cos-map 3 5 7
! Assigns Network-Control (Spanning Tree) to Q3 WRED T5
C7600(config-if)# priority-queue cos-map 1 5
! Assigns VoIP to the PQ (Q4)
C7600(config-if)#end
C7600#
```

Cisco 7600 1P7Q8T 10GE Queuing Design

The Cisco 7600 1P7Q8T queuing structure adds four additional standard queues to the 1P3Q8T structure and moves the PQ from Q4 to Q8, but otherwise is identical.

Under a 1P7Q8T model, buffer space can be allocated as follows:

- 5 percent for the Scavenger/Bulk queue (Q1)
- 25 percent for the Best Effort queue (Q2)
- 10 percent for the Video queue (Q3)
- 10 percent for the Network-Management/Transactional Data queue (Q4)
- 10 percent for the Call-Signaling/Mission-Critical Data queue (Q5)
- 5 percent for the Internetwork-Control queue (Q6)
- 5 percent for the Network Control queue (Q7)
- 30 percent for the PQ (Q8)

The WRR weights for the standard queues (Q1 through Q7) for dividing the remaining bandwidth after the priority queue has been fully serviced can be set to 5:25:20:20:20:5:5 respectively for Q1 through Q7.

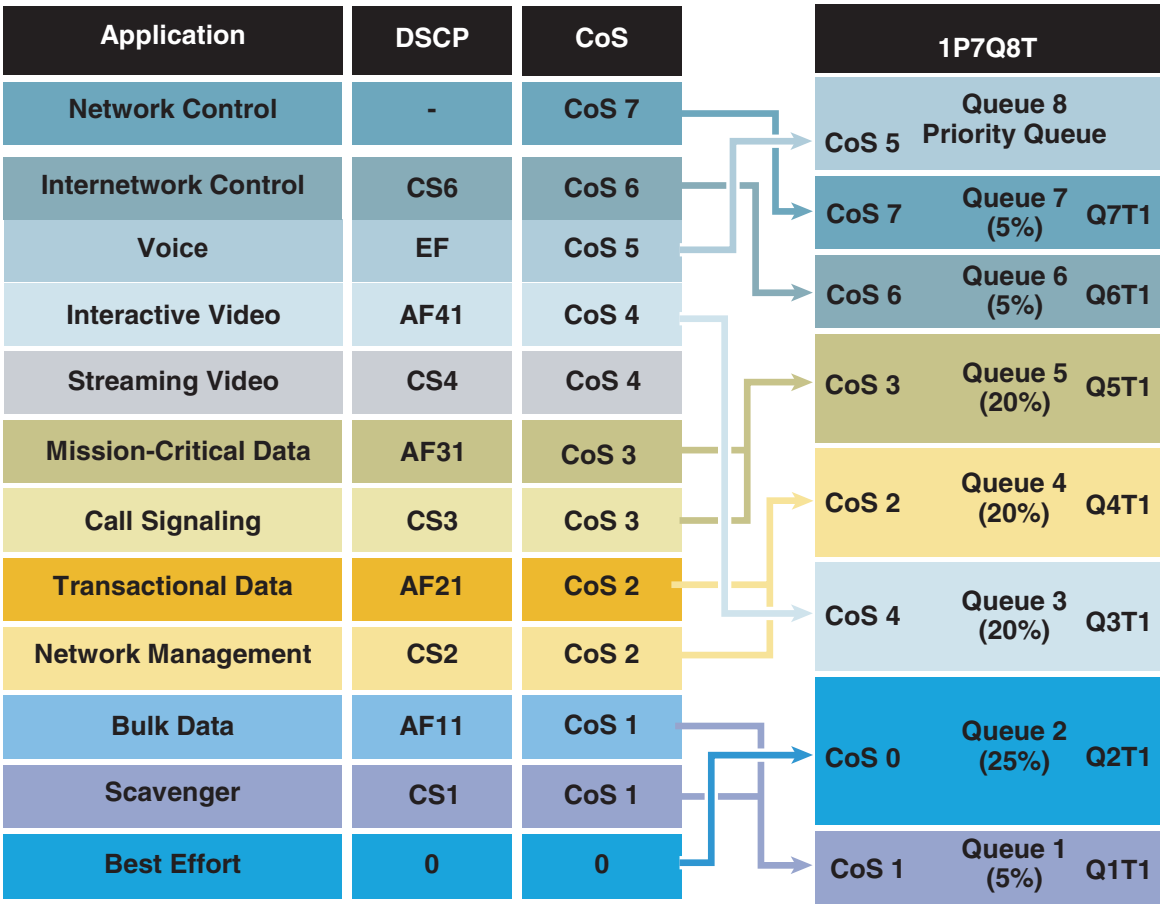
Because eight queues are available, each CoS value can be assigned to its own exclusive queue. WRED can be enabled on each queue to provide it with congestion avoidance by setting the first WRED threshold of each queue to 80:100. All other WRED thresholds can remain at 100:100.

After the queues and thresholds have been defined as above, the following assignments can be made:

- CoS 1 (Scavenger/Bulk) to Q1T1
- CoS 0 (Best Effort) to Q2T1
- CoS 4 (Interactive and Streaming Video) to Q3T1
- CoS 2 (Network Management and Transactional Data) to Q4T1
- CoS 3 (Call-Signaling and Mission-Critical Data) to Q5T1
- CoS 6 (Internetwork Control) to Q6T1
- CoS 7 (Internetwork and Network Control) to Q7T1
- CoS 5 (VoIP) to Q8 (the PQ)

These 1P7Q8T queuing recommendations are illustrated in Figure A-11.

Figure A-11 Cisco 7600 1P7Q8T Queuing Model



The Cisco 7600 commands configure 1P7Q8T queuing recommendations are shown in the following configuration example.

```
C7600(config)#interface range TenGigabitEthernet4/1 - 4
C7600(config-if-range)# wrr-queue queue-limit 5 25 10 10 10 5 5
! Allocates 5% to Q1, 25% to Q2, 10% to Q3, 10% to Q4,
! Allocates 10% to Q5, 5% to Q6 and 5% to Q7
C7600(config-if-range)# wrr-queue bandwidth 5 25 20 20 20 5 5
! Sets the WRR weights for 5:25:20:20:20:5:5 (Q1 through Q7)
C7600(config-if-range)#
```

```
C7600(config-if-range)#
C7600(config-if-range)# wrr-queue random-detect 1
! Enables WRED on Q1
C7600(config-if-range)# wrr-queue random-detect 2
! Enables WRED on Q2
C7600(config-if-range)# wrr-queue random-detect 3
! Enables WRED on Q3
C7600(config-if-range)# wrr-queue random-detect 4
! Enables WRED on Q4
C7600(config-if-range)# wrr-queue random-detect 5
! Enables WRED on Q5
C7600(config-if-range)# wrr-queue random-detect 6
! Enables WRED on Q6
C7600(config-if-range)# wrr-queue random-detect 7
! Enables WRED on Q7
C7600(config-if-range)#
C7600(config-if-range)#
C7600(config-if-range)# wrr-queue random-detect min-threshold 1 80 100 100 100 100 100 100
100
! Sets Min WRED Threshold for Q1T1 to 80% and all others to 100%
C7600(config-if-range)# wrr-queue random-detect max-threshold 1 100 100 100 100 100 100
100 100
! Sets Max WRED Threshold for Q1T1 to 100% and all others to 100%
C7600(config-if-range)#
C7600(config-if-range)# wrr-queue random-detect min-threshold 2 80 100 100 100 100 100 100
100
! Sets Min WRED Threshold for Q2T1 to 80% and all others to 100%
C7600(config-if-range)# wrr-queue random-detect max-threshold 2 100 100 100 100 100 100
100 100
! Sets Max WRED Threshold for Q2T1 to 100% and all others to 100%
C7600(config-if-range)#
C7600(config-if-range)# wrr-queue random-detect min-threshold 3 80 100 100 100 100 100 100
100
! Sets Min WRED Threshold for Q3T1 to 80% and all others to 100%
C7600(config-if-range)# wrr-queue random-detect max-threshold 3 100 100 100 100 100 100
100 100
! Sets Max WRED Threshold for Q3T1 to 100% and all others to 100%
C7600(config-if-range)#
C7600(config-if-range)# wrr-queue random-detect min-threshold 4 80 100 100 100 100 100 100
100
! Sets Min WRED Threshold for Q4T1 to 80% and all others to 100%
C7600(config-if-range)# wrr-queue random-detect max-threshold 4 100 100 100 100 100 100
100 100
! Sets Max WRED Threshold for Q4T1 to 100% and all others to 100%
C7600(config-if-range)#
C7600(config-if-range)# wrr-queue random-detect min-threshold 5 80 100 100 100 100 100 100
100
! Sets Min WRED Threshold for Q5T1 to 80% and all others to 100%
C7600(config-if-range)# wrr-queue random-detect max-threshold 5 100 100 100 100 100 100
100 100
! Sets Max WRED Threshold for Q5T1 to 100% and all others to 100%
C7600(config-if-range)#
C7600(config-if-range)# wrr-queue random-detect min-threshold 6 80 100 100 100 100 100 100
100
! Sets Min WRED Threshold for Q6T1 to 80% and all others to 100%
C7600(config-if-range)# wrr-queue random-detect max-threshold 6 100 100 100 100 100 100
100 100
! Sets Max WRED Threshold for Q6T1 to 100% and all others to 100%
C7600(config-if-range)#
C7600(config-if-range)# wrr-queue random-detect min-threshold 7 80 100 100 100 100 100 100
100
! Sets Min WRED Threshold for Q7T1 to 80% and all others to 100%
C7600(config-if-range)# wrr-queue random-detect max-threshold 7 100 100 100 100 100 100
100 100
```

```
! Sets Max WRED Threshold for Q7T1 to 100% and all others to 100%
C7600(config-if-range)#
C7600(config-if-range)#
C7600(config-if-range)# wrr-queue cos-map 1 1 1
! Assigns Scavenger/Bulk to Q1 WRED Threshold 1
C7600(config-if-range)# wrr-queue cos-map 2 1 0
! Assigns Best Effort to Q2 WRED Threshold 1
C7600(config-if-range)# wrr-queue cos-map 3 1 4
! Assigns Video to Q3 WRED Threshold 1
C7600(config-if-range)# wrr-queue cos-map 4 1 2
! Assigns Net-Mgmt and Transactional Data to Q4 WRED T1
C7600(config-if-range)# wrr-queue cos-map 5 1 3
! Assigns call signaling and Mission-Critical Data to Q5 WRED T1
C7600(config-if-range)# wrr-queue cos-map 6 1 6
! Assigns Internetwork-Control (IP Routing) to Q6 WRED T1
C7600(config-if-range)# wrr-queue cos-map 7 1 7
! Assigns Network-Control (Spanning Tree) to Q7 WRED T1
C7600(config-if-range)# priority-queue cos-map 1 5
! Assigns VoIP to the PQ (Q4)
C7600(config-if-range)#end
C7600-IOS#
```

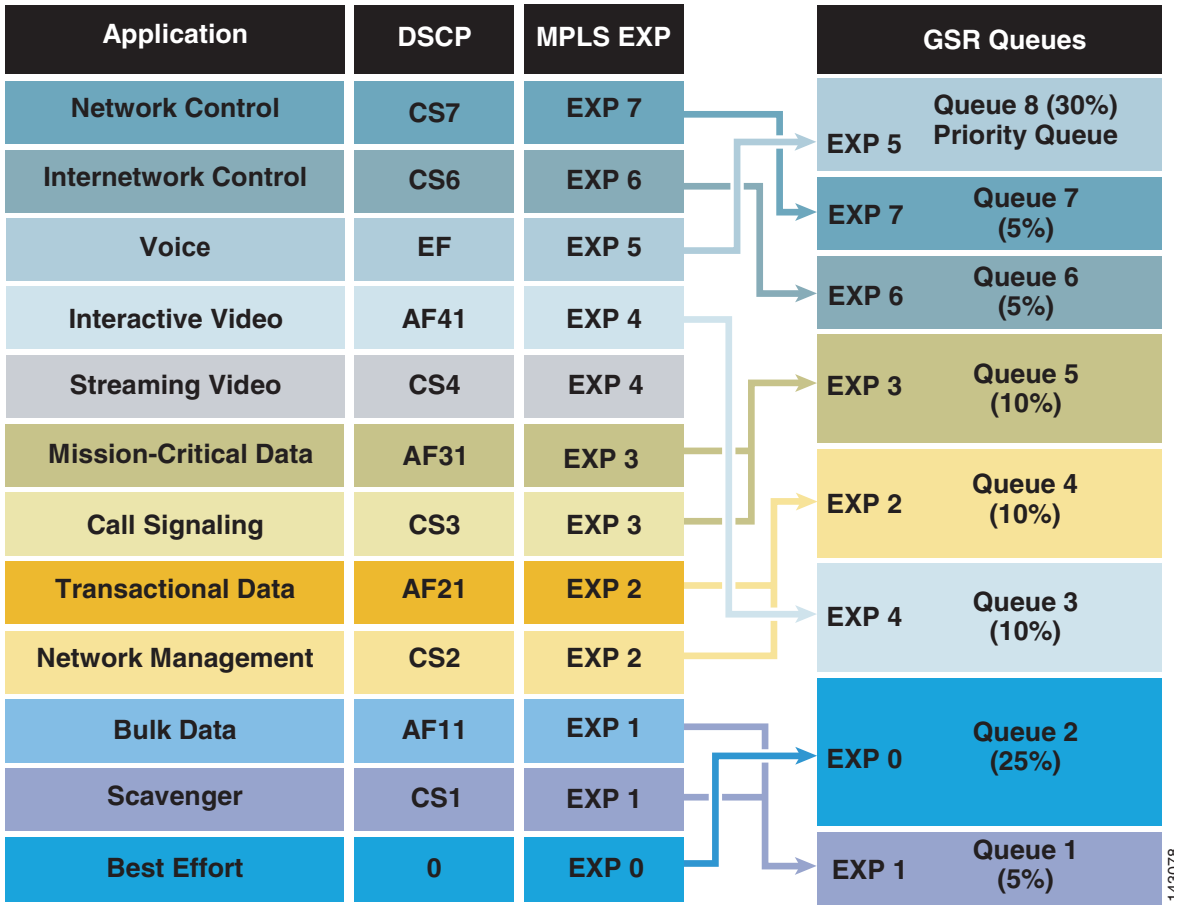
Cisco 12000 QoS Design

Cisco 12000 series routers are also an option as edge or core routers in the NG-WAN/MAN. These high performance routers have long been deployed in performance intensive networks of service providers and enterprises and offer extremely rich features with high performance. This section describes the specific QoS considerations when Cisco 12000 series routers is used in enterprise networks as the PE and the P routers using ISE (or Engine 5) Gigabit line cards.

Cisco 12000 GSR Edge Configuration

MPLS EXP is three bits in length and can support a maximum of eight traffic classes. As mentioned earlier, you assume the maximum of 8 classes in the core. The 11 enterprise traffic classes are mapped to the 8 core classes at the ingress PE. Although not more than three traffic classes are typical in a service provider core, an enterprise core can have up to 8 traffic classes to provide better control over the individual classes as illustrated in Figure A-12.

Figure A-12 Cisco 12000 8-Class Queuing Model

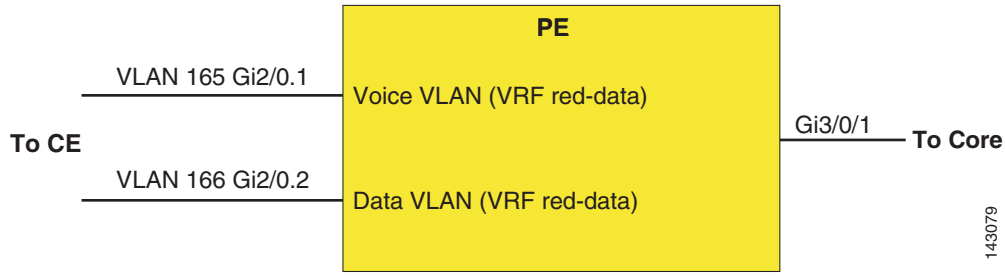


The QoS features of ISE (Engine 3) 4-port Gigabit Ethernet line card makes it very suitable for being used as an edge line card although it can also be used in the core as well. The following sections are based on using this line card as an edge line card. Engine 5 line cards, being QoS compatible with the Engine 3 line cards, can be used instead. Cisco IOS 12.0.30S2 or above is assumed.

The simplest QoS configuration at the ingress PE is to assume that all traffic classes are processed by the main interface without having any subinterface. In this case a single service policy is attached to the main interface (or sub-interface).

However some enterprises may need to segregate their Voice and other traffic into separate VRFs. In this case you can send the traffic via different VLANs terminating at separate subinterfaces. These subinterfaces are mapped to different VRFs. The service policy is still attached to the main interface as shown in Figure A-13.

Figure A-13 Sample PE Configuration with Separate VRFs for Voice and other Data Traffic



PE Config (CE Facing Configuration—Ingress QoS)

No specific ingress QoS is configured for policing or marking in this example. Because the PE router is not separating QoS domains with different marking policies, no packet remarking is necessary. Further it is assumed that the default mapping of IP Precedence (or DSCP) to MPLS EXP is used (although it is fine to have a policy map use a different mapping on ingress, if so desired).

An enterprise may not enforce any rate limits of different traffic classes. In case rate-limiting is a requirement, an appropriate service-policy using policing can be attached to the individual sub-interfaces.

PE Config (CE Facing Configuration—Egress QoS)

```
interface GigabitEthernet2/0
 description To DL2 - intf G5/2 - CE facing
 no ip address
 no ip directed-broadcast
 negotiation auto
 service-policy output q-2ce-out-parent
!
interface GigabitEthernet2/0.1
 description RED-DATA
 encapsulation dot1Q 165
 ip vrf forwarding red-data
 ip address 125.1.102.49 255.255.255.252
 no ip directed-broadcast
 ip pim sparse-mode
!
interface GigabitEthernet2/0.2
 description RED-VOICE
 encapsulation dot1Q 166
 ip vrf forwarding red-voice
 ip address 125.1.102.53 255.255.255.252
 no ip directed-broadcast
 ip pim sparse-mode

class-map match-all red-voice   <------- VLAN 166 carries Voice Traffic (plus routing
traffic)
  match vlan  166
class-map match-all red-data   < -------- VLAN 165 carries rest of the Traffic classes (+
routing)
  match vlan  165
```

```
class-map match-any realtime-2ce
  match qos-group 5
class-map match-any network-control-2ce
  match qos-group 7
class-map match-any bulk-data-2ce
  match qos-group 1
class-map match-any interwork-control-2ce
  match qos-group 6
  match  IP precedence 6
class-map match-any bus-critical-2ce
  match qos-group 3
class-map match-any trans-data-2ce
  match qos-group 2
class-map match-any video-2ce
  match qos-group 4



policy-map q-2ce-out-parent  <-------- Policy map attached to the main interface
  class red-data           <------- No Priority traffic, but seven classes of traffic,
plus OSPF
    shape average percent 50
    service-policy q-2ce-out-1
  class red-voice                       <----- Carries voice traffic + OSPF
    shape average percent 40
    service-policy q-2ce-out-2

The child policy for voice VRF:

policy-map q-2ce-out-2
  class realtime-2ce
    priority
   police cir percent 95 bc 500 ms conform-action transmit  exceed-action drop
  class interwork-control-2ce     <== OSPF
    bandwidth percent 3
    random-detect
    random-detect precedence 6 4720 packets 4721 packets 1
  class class-default
    bandwidth percent 2
    random-detect                          <===  There is no traffic in this class
    random-detect precedence 6 4720 packets 4721 packets 1
```

Note If a traffic class is configured with a "priority" command using MQC, important traffic marked with PAK_PRIORITY (for example, routing traffic) goes either to the class matching IP Precedence 6 if defined or to a class matching IP Precedence 7 if defined or to the "class-default." Because in the above case a class with the "priority" command is defined, OSPF traffic to the CE goes to the interwork-control-2ce class that matches IP Precedence 6.

The child policy map for data VRF is as follows:

```
policy-map q-2ce-out-1           <--- DATA VLAN policy (no priority Q)
  class network-control-2ce
    bandwidth percent 7
    random-detect discard-class-based
    random-detect discard-class 7 625 packets 4721 packets 1
  class interwork-control-2ce
    bandwidth percent 7
    random-detect discard-class-based
    random-detect discard-class 6 625 packets 4721 packets 1
  class bus-critical-2ce
    bandwidth percent 14
```

```
    random-detect discard-class-based
    random-detect discard-class 3 625 packets 4721 packets 1
  class trans-data-2ce
    bandwidth percent 14
    random-detect discard-class-based
    random-detect discard-class 2 625 packets 4721 packets 1
  class video-2ce
    bandwidth percent 14
    random-detect discard-class-based
    random-detect discard-class 4 625 packets 4721 packets 1
  class bulk-data-2ce
    bandwidth percent 7
    random-detect discard-class-based
    random-detect discard-class 1 625 packets 4721 packets 1
  class class-default
    bandwidth percent 36
    random-detect discard-class-based
    random-detect discard-class 0 625 packets 4721 packets 1
```

Note If a traffic class is *not* configured with a "priority" command using MQC (as in the above case, routing traffic goes either to the class matching IP Precedence 6 if defined or to a class matching IP Precedence 7 if defined or to the priority queue. Because in the above case there is *not* defined a class with the "priority" command, OSPF traffic to the CE goes to the priority queue.

```
PE Config (P Facing Configuration – Egress QoS)
class-map match-any realtime
  match mpls experimental  5
class-map match-any bulk-data
  match mpls experimental  1
class-map match-any interwork-control
  match mpls experimental  6
class-map match-any network-control
  match mpls experimental  7
class-map match-any bus-critical
  match mpls experimental  3
class-map match-any trans-data
  match mpls experimental  2
class-map match-any video
  match mpls experimental  4

policy-map q-core-out
  class realtime
    priority
   police cir percent 30 bc 500 ms conform-action transmit  exceed-action drop
  class network-control
    bandwidth remaining percent 7
    random-detect
    random-detect precedence 7 625 packets 4721 packets 1
  class interwork-control
    bandwidth remaining percent 7
    random-detect
    random-detect precedence 6 625 packets 4721 packets 1
  class bus-critical
    bandwidth remaining percent 14
    random-detect
    random-detect precedence 3 625 packets 4721 packets 1
  class trans-data
    bandwidth remaining percent 14
    random-detect
    random-detect precedence 2 625 packets 4721 packets 1
  class video
    bandwidth remaining percent 14
```

```
    random-detect
    random-detect precedence 4 625 packets 4721 packets 1
  class bulk-data
    bandwidth remaining percent 7
    random-detect
    random-detect precedence 1 625 packets 4721 packets 1
  class class-default
    bandwidth remaining percent 36
    random-detect
    random-detect precedence 0 625 packets 4721 packets 1

interface GigabitEthernet3/3/0
 description To P1 - intf G4/0/2
 ip address 125.1.102.6 255.255.255.252
 no ip directed-broadcast
 ip pim sparse-mode
 load-interval 30
 negotiation auto
 tag-switching ip
 service-policy input egr-pe-in
 service-policy output q-core-out
```

PE Config (P Facing Configuration—Ingress QoS)

```
class-map match-any realtime
  match mpls experimental  5
class-map match-any bulk-data
  match mpls experimental  1
class-map match-any interwork-control
  match mpls experimental  6
class-map match-any network-control
  match mpls experimental  7
class-map match-any bus-critical
  match mpls experimental  3
class-map match-any trans-data
  match mpls experimental  2
class-map match-any video
  match mpls experimental  4

policy-map egr-pe-in
  class realtime
   set qos-group 5
   set discard-class 5
  class network-control
   set qos-group 7
   set discard-class 7
  class interwork-control
   set qos-group 6
   set discard-class 6
  class bus-critical
   set qos-group 3
   set discard-class 3
  class trans-data
   set qos-group 2
   set discard-class 2
  class video
   set qos-group 4
   set discard-class 4
  class bulk-data
   set qos-group 1
   set discard-class 1
```

The following are general notes on the ISE 4-Port GigabitEthernet line card:

- This line card supports only four traffic classes by default. However it can support up to eight traffic classes when you configure the following:

  ```
  hw-module slot <slot#> qos interface queues 8
  ```

- The MQC bandwidth, shape commands, and policers by default use Layer 3 packet size for bandwidth calculations. If you want to include Layer 2 header size in bandwidth calculations, use the following command:

  ```
  hw-module slot slot-number qos-account-layer2-encapsulation {arpa | dot1q | length}
  ```

Cisco 12000 GSR ToFab Queuing

Typically, no specific QoS configuration is necessary for incoming traffic at the ingress interface. However, in a GSR, "line card-to-fabric" queuing (or toFab queuing) should be configured to avoid congestion of traffic from the line cards to the switching fabric. Such congestion may happen when multiple ports on the ingress line card receive incoming traffic peaks at the exact same time and the sum of the peak bandwidths exceed the fabric bandwidth.

WRED is less important here because WRED works on sustained congestion, while the ToFab congestion is likely to be of instantaneous nature. ToFab queuing on a GSR is configured using legacy CLI as in the following configuration.

Although the following sample configuration is for three core classes, it can be expanded accordingly when eight core classes are used.

```
slot-table-cos SLOT_TABLE
 destination-slot all core_policy
!
rx-cos-slot all SLOT_TABLE
!
cos-queue-group core_policy
  precedence 0 queue 0 ! Traffic with IP Prec 0 or EXP 0 go to queue # 0 (best  effort)
  precedence 1 queue 1
  precedence 2 queue 1 ! Traffic with IP Prec(EXP) 2,3,6,1 go to queue # 1 (businessclass)
  precedence 3 queue 1
  precedence 6 queue 1
  precedence 5 queue low-latency ! Traffic with IP Precedence 5 or EXP 5 go to PQ (real
time class)
  precedence 0 random-detect-label 1       ! For traffic with precedence 0, use WRED label
number 1
precedence 1 random-detect-label 0       ! For traffic with precedence 0, use WRED label
number 1
                                                                  ! out-of-contract
traffic, so lower thresholds
  precedence 2 random-detect-label 1
  precedence 3 random-detect-label 1
  precedence 6 random-detect-label 1
random-detect-label 0   500 1012 1          ! WRED label # 0 defines the WRED thresholds

random-detect-label 1 1500 9692 1
  queue 0 1                                                    ! Queue weight
  queue 1 71                                                  ! Queue weight
  queue low-latency strict                              ! Configures strict priority to real
time traffic class
```

Note the following:

- The above configuration is GSR specific and in native GSR CLI.

- The toFab queuing uses MDRR algorithm where you assign relative weights to the non- priority queues. The weights are used to calculate the quantum of a queue (maximum number of bytes that can be output from the queue in one round-robin cycle). The ratios of the quanta among the queues determine how the excess bandwidth is allocated to these queues after the PQ has been serviced.

  The Quantum for a queue is calculated as follows:

  Quantum = MTU + (queue weight - 1) * 512
- Although the configuration specifies IP Precedence, it also matches EXP for MPLS packets.
- See the next section for WRED parameter calculation

WRED Tuning at the Edge and Core

In general, WRED tuning is a complex task and depends on several factors such as traffic load, traffic mix, the ratio of the offered load to the link capacity, and traffic behavior in the presence of congestion (for example, how do the TCP stacks react to traffic drops may vary among implementations). These factors vary from network to network and depend on the type of services offered as well as the properties of the applications such as TCP stacks the customers run. It is therefore very difficult to provide recommendations that work equally well in all networks.

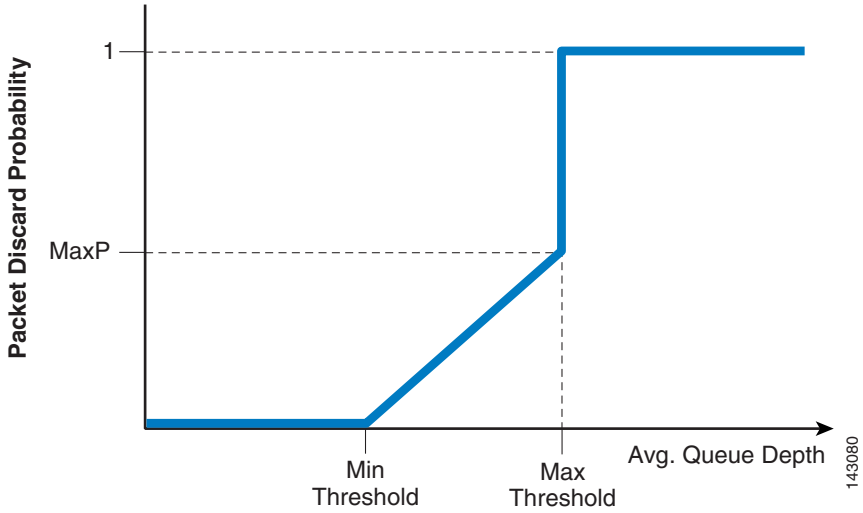
The recommendations given here should be taken as a starting point and should be further refined per real life testing and operational experience in the target network environment.

Without proper parameter setting, WRED may affect link utilization and latency. If the WRED thresholds are too large, a packet near the tail end of the queue can spend a longer time before being scheduled for transmission, thus increasing latency. On the other hand, having a small WRED queue may drop more packets, thus decreasing link utilization. WRED tuning involves setting WRED parameters such as the minimum, maximum thresholds, and drop probabilities so that the link utilization is maximized, while the mean WRED queue depth per class is minimized to decrease latency.

Typical WRED parameters that can be tuned are the minimum and maximum thresholds, the drop probability, and the exponential weighting constant. It is recommended to set the drop probability to 1 and to not change the default value of the exponential weighting constant. Tuning of the other parameters (min and max threshold) depends on the traffic volume, RTT (round trip time), and interface MTU (assumed to be 1500 for Ethernet).

Figure A-14 shows WRED parameters that can be tuned.

Figure A-14 WRED Parameters for Tuning



The WRED tuning goals and how they can be realized are described as follows for link speeds of 10 Mbps or higher:

- Min threshold should be high enough so that link utilization is maximized.

  Set min threshold = 0.15 * P, where

  P is the pipe size = RTT * BW / (MTU * 8)

  RTT = Round Trip Time ~ 100 ms (typical value used in USA)

  MTU = Maximum Transmission Unit of the interface (use 1500 as MTU in this formula, even if the actual MTU is configured on the interface is higher; for example, 4470)

- The difference between the min and the max threshold should be high enough to avoid TCP global synchronization.

  If the difference is small, then too many packets could be dropped in a very small time interval (which would be nearly similar to a tail-drop situation) leading to TCP global synchronization and consequent throughput reduction.

  The recommendation for max threshold is:

  Max threshold = 1 * P

The other rules that apply while calculating these parameters are as follows:

- The difference between the max and the min threshold should be a power of two (for GSR routers).
- Min and max threshold should be calculated based on the traffic volume on the link, rather than the full link speed. For example, assuming that traffic, on an average, would not exceed 50 percent of the link speed, the min and max thresholds would be half of the calculated values by the above formulas. Lower traffic volume may allow lower thresholds.
- Set the packet discard probability to 1; that is, drop all exceeding packets of a class after the WRED max threshold is exceeded.
- Do not change exponential weighting constant from its default.

With the above rules, min and max thresholds for an OC-48 link (2.5 Gbps) are calculated as follows:

P = 100ms * 2.5Gbps / (1500 * 8) = 20,000

Therefore, min threshold = 0.15 * 20,000 = 3000, and max threshold = 20000

Assuming 50 percent normal traffic load, min and max thresholds are adjusted to 1500 and 10000 respectively. However because their difference should be a power of 2, you further adjust the max threshold to 9692.

Similar calculations for OC-3, OC-12, and Gigabit Ethernet are summarized in Table A-3:

Table A-3 Minimum and Maximum Thresholds

Link Speed	Pipe Size (P)	Minimum Threshold	Maximum Threshold
OC-3	1292	97	609
OC-12	5184	389	2437
OC-48	20000	1500	9692
Gigabit		625	4176

Note that these values are to be taken as starting values. These assume 50 percent link rate utilization for the traffic class and can be further adjusted per the actual utilization for the classes and also per real life traffic pattern.

APPENDIX B

Terminology

- Provider (P) network—P network is the backbone under the control of the enterprise IT.
- Provider edge (PE) interface—The PE interface is located at the edge of a provider network and faces the customer.
- Provider edge (PE) router—A router belonging to a provider with one or more PE interfaces; also known as an edge label switching router (ELSR). PE routers connect to VPN site routers to provide connectivity into an MPLS network.
- Provider (P) router—A router belonging to a provider with no PE interfaces. P routers reside in the core of the service provider and provide interconnectivity to PE routers.
- Customer edge (CE) interface—An interface on a customer router that points towards a provider, where customer router means a router that is logically part of the customer network no matter where it is physically located or who manages it.
- Customer edge (CE) router—A router with at least one customer edge interface that resides on a customer premise. In a typical campus design, access layer switches are Layer 2-attached to the distribution layer. Access layer VLANs terminate on the first Layer 3 device (the distribution layer), so CE router terminology does not apply to campus networks. Consider the CE box as a Layer 2 switch connected to the distribution layer router.
- Multi-VRF (CE) router—This router supports VRFs but not full MP-BGP VPNs. This could reside on a customer premise. In a campus network, this could be any Layer 3 switch where VLANs terminate.