Avaya IP Telephony Implementation Guide

Communication Manager 3.1 Avaya Labs

ABSTRACT

This document gives implementation guidelines for the Avaya MultiVantage™ Communications Applications. Configurations and recommendations are given for various Avaya Media Servers and Gateways, as well as Avaya 4600 Series IP Telephones. This document also provides information on virtual local area networks (VLANs), and guidelines for configuring Avaya and Cisco networking equipment in VoIP applications.

The intent of this document is to provide training on IP telephony, and to give guidelines for implementing Avaya solutions. It is intended to supplement the product documentation, not replace it. This document covers the Avaya Communication Manager 2.2 through 3.1, and the Avaya 4600 Series IP Telephone 1.8 and later, with limited information regarding previous and future versions.

External posting: www.avaya.com.

May 2006
COMPAS ID 95180

All information in this document is subject to change without notice. Although the information is believed to be accurate, it is provided without guarantee of complete accuracy and without warranty of any kind. It is the user's responsibility to verify and test all information in this document. Avaya shall not be liable for any adverse outcomes resulting from the application of this document; the user accepts full responsibility.
The instructions and tests in this document regarding Cisco products and features are best-effort attempts at summarizing and testing the information and advertised features that are openly available at www.cisco.com. Although all reasonable efforts have been made to provide accurate information regarding Cisco products and features, Avaya makes no claim of complete accuracy and shall not be liable for adverse outcomes resulting from discrepancies. It is the user's responsibility to verify and test all information in this document related to Cisco products, and the user accepts full responsibility for all resulting outcomes.

Foreword

Several benefits are motivating companies to transmit voice communications over packet networks originally designed for data.

Cost saving is one factor. By eliminating a separate circuit-switched voice network, businesses avoid the expenses of buying, maintaining and administering two networks. They may also reduce toll charges by sending long distance voice traffic over the enterprise network, rather than the public switched telephone network.

Another benefit is the potential to more tightly integrate data and voice applications. Because they use open programming standards, Avaya MultiVantage™ products make it easier for developers to create, and for companies to implement, applications that combine the power of voice and data in such areas as customer relationship management (CRM) and unified communications. A converged multi-service network can make such applications available to every employee.

These benefits do not come free, however. Voice and data communications place distinctly different demands on the network. Voice and video are real-time communications that require immediate transmission. Data does not. Performance characteristics that work fine for data can produce entirely unsatisfactory results for voice or video. So networks that transmit all three must be managed to meet the disparate requirements of data and voice/video.

Network managers are implementing a range of techniques to help ensure their converged networks meet performance standards for all three payloads: voice, video and data. These techniques include the strategic placement of VLANs, and the use of Class of Service (CoS) packet marking and Quality of Service (QoS) network mechanisms.

For an overview of IP telephony issues and networking requirements, see the "Avaya IP Voice Quality Network Requirements" white paper.

Professional consulting services are available through the Avaya Communication Solutions and Integration (CSI) group. One essential function of this professional services group is to provide pre-deployment network assessments to Avaya customers. This assessment helps to prepare a customer's network for IP telephony, and also gives critical network information to Avaya support groups that will later assist with implementation and troubleshooting. Arrange for this essential service through an Avaya account team.

Avaya IP Telephony Implementation Guide

Table of Contents

1 Introduction to VoIP and Avaya Products

This section provides a foundation to build upon for the rest of this document. Voice over IP (VoIP) terminology and Avaya products and architectures are introduced here.

1.1 Servers, Gateways, Stations, and Trunks Defined

Servers

Most of the intelligence in a voice system is in the call server. From servicing a simple call to making complex decisions associated with large contact centers, the call server is the primary component of an IP telephony system. Avaya Communication Manager is the call processing software that runs on Avaya server platforms.
The following are some common terms for a call server. Some are generic and some are specified by a protocol, but all are generally used throughout the industry.

- Call Server – generic term
- Call Controller – generic term
- Gatekeeper – H.323 term
- Media Gateway Controller – H.248 term

Gateways

A gateway terminates and converts various media types, such as analog, TDM, and IP. A gateway is required so that, for example, an IP phone can communicate with an analog phone on the same telephony system, as well as with an external caller across a TDM trunk.
The following are some common terms for a gateway, and they are generally used throughout the industry.

- Gateway – generic and H.323 term
- Media Gateway – H.248 term
- Port Network – Avaya term

A gateway requires a call server to function, and some common Avaya server-gateway architectures are illustrated later.

Stations

There are several technical terms for what most would call a telephone, and some that are generally used throughout the industry are listed below.

- Endpoint – H.323 general term that includes IP phones and other endpoints
- Terminal – H.323 specific term to mean primarily IP phones (also an Avaya term)
- Station – Avaya term, and possibly a generic term
- Set – Avaya term, and possibly a generic term

Avaya gateways have port boards or media modules that terminate various types of stations.

Trunks

Trunks connect independent telephony systems together, such as PBX to PBX, or PBX to public switch, or public switch to public switch. In traditional telephony there are various types of circuit-switched trunks, using various protocols to signal across these trunks. IP telephony introduces another trunk type – the IP trunk. Like circuit-switched trunks IP trunks connect independent telephony systems together, but the medium is an IP network and the upper-layer protocol suite is H.323.
Avaya gateways have port boards or media modules that terminate various types of trunks, including IP trunks.

1.2 Avaya Server-Gateway and Trunk Architectures

The following figures illustrate some common Avaya server-gateway architectures in succession, from established to most recent technologies. Also included in the diagrams are the protocols used to communicate between the various devices.

Traditional DEFINITY® System

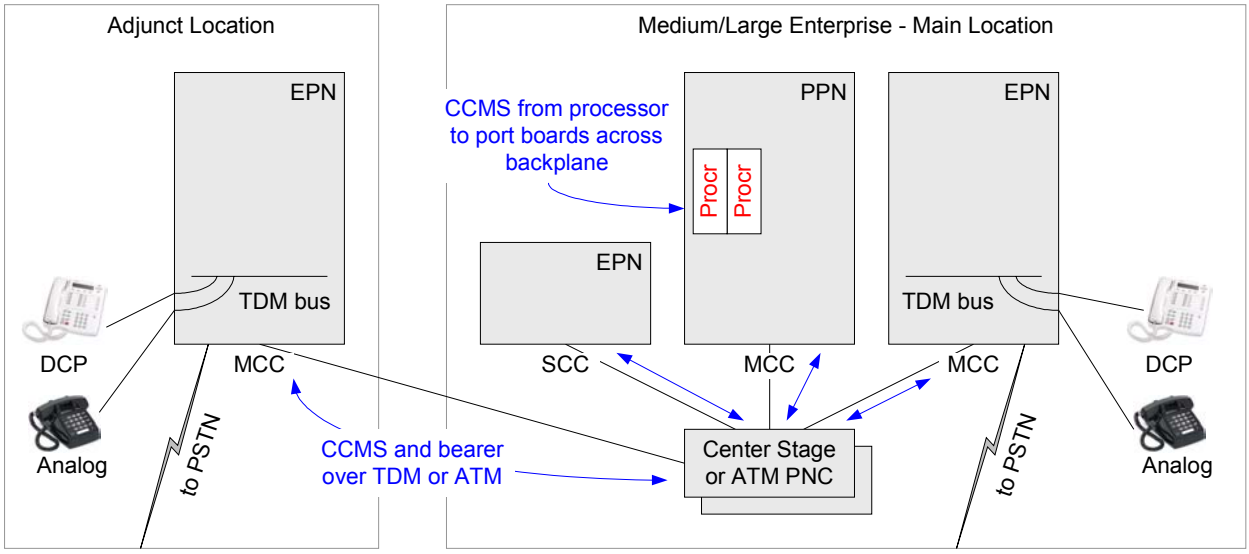


Figure 1: Traditional DEFINITY System architecture

- The single- (SCC1) and multi-carrier cabinets (MCC1) are called port networks (Avaya term) or media gateways (VoIP term). They house port boards, which, among other things, terminate stations and trunks. (These port boards are not the focus of this document.)
- The DEFINITY® call servers are the processor boards inserted into the processor port network (PPN).
- The other cabinets, without processors, are called expansion port networks (EPN) and are controlled by the DEFINITY servers in the PPN.
- The port networks are connected together via a port network connectivity (PNC) solution, which can be TDM-based (Center Stage PNC) or ATM-based (ATM PNC). Both bearer (audio) and port network control are carried across the PNC solutions.
- Control Channel Message Set (CCMS) is the Avaya proprietary protocol used by the DEFINITY servers to control the port networks (cabinets and port boards).

IP-enabled DEFINITY System

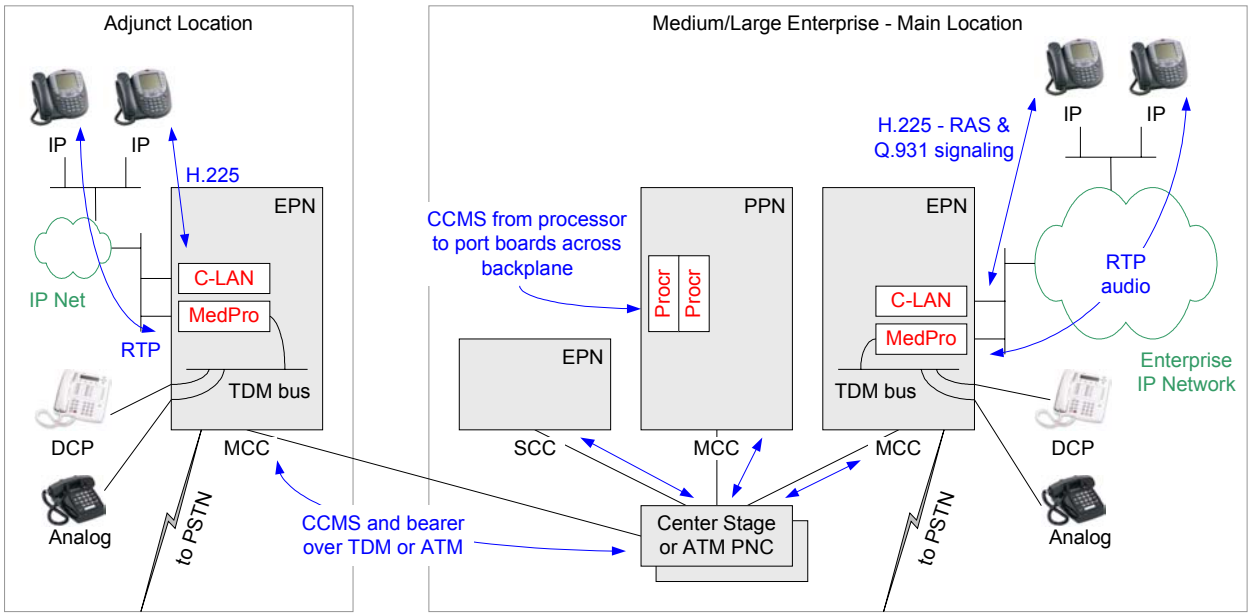


Figure 2: IP-enabled DEFINITY System

- IP-enabled DEFINITY System is the same architecture as before, but with IP port boards added.
- The Control-LAN (C-LAN) board is the call servers' interface into the IP network for call signaling. H.225, which is a component of H.323, is the protocol used for call signaling. H.225 itself has two components: RAS for endpoint registration, and Q.931 for call signaling.
- The IP Media Processor (MedPro) board is the IP termination point for audio. As of Communication Manager 3.0 there is a higher capacity version of the MedPro board called IP Media Resource 320 (MR320). The MR320 supports up to 320 calls, based on licensing, as opposed to a fixed max of 64 calls for the MedPro. These boards perform the conversion between TDM and IP. The audio payload is encapsulated in RTP, then UDP, then IP.

Multi-Connect

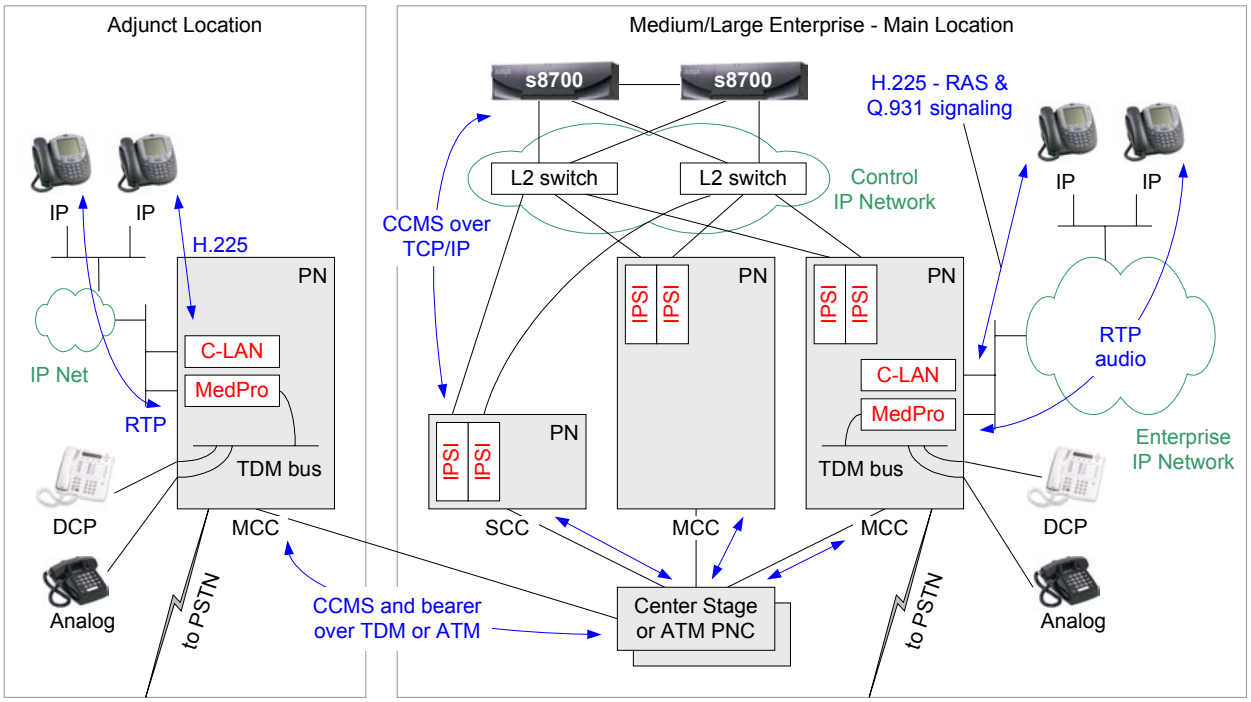


Figure 3: Multi-Connect

- Multi-Connect is the same underlying DEFINITY architecture, except that the processor boards are replaced with much more powerful Avaya S8700 or S8710 Media Servers.
- Port networks get IP Server Interface (IPSI) boards to communicate with the S87xx call servers.
- CCMS exchanges between the call servers and port networks now take place over the control IP network.
- Not all port networks require IPSI boards, because Center Stage PNC and ATM PNC are still present to connect the port networks.

S8500 Media Server

Figure 4: Avaya S8500 Media Server

The Avaya S8500 Media Server is the simplex equivalent of the S87xx server pair. The S8500 gives the same call processing capability without the redundancy and added reliability of duplicated servers. The S8500 can be substituted in place of the S87xx servers in any IP-Connect configuration shown below.

IP-Connect

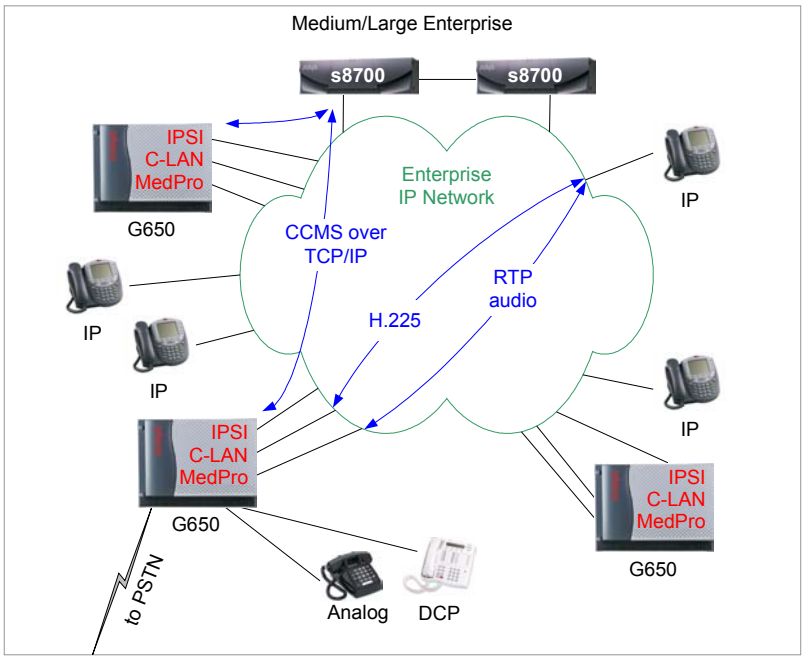


Figure 5: IP-Connect

- With IP-Connect the traditional port networks – MCC1 and SCC1 – are replaced with new, 19-inch rack-mountable Avaya G650 or G600 Media Gateways.
- All G650/G600s require IPSI boards; no more Center Stage or ATM PNC.
- Everything is done on the enterprise IP network; no more control IP network.
- G650/G600 media gateways still use C-LAN and MedPro/MR320 boards, as well as the other traditional port boards used in the MCC1 and SCC1.

S8300/G700/G350/G250

- The Avaya G700, G350, and G250 (not shown) Media Gateways are based on the H.248 protocol.
- One primary difference between these gateways is capacity. (Refer to current product offerings for exact specifications.)
- All gateways have built-in Ethernet switches. The G700 supports IP routing and IP WAN connectivity with an expansion module, and the G350 and G250 support them natively.
- The G700 is built on the Avaya P330 Stackable Switching System, with similar CLI. The G350 and G250 are built on a new IP platform, also with similar CLI.
- All gateways use compact media modules instead of traditional port boards.
- The VoIP media module serves the same function as the MedPro board.
- There are other media modules equivalent to traditional port boards (analog, DCP, DS1).
- The Avaya S8300 Media Server in internal call controller (ICC) mode is the call server.
- The S8300 is a Linux platform, similar to the S8700, but in a compact form factor that fits into these gateways.
- The S8300 is not front-ended by C-LANs; it terminates the call signaling natively.

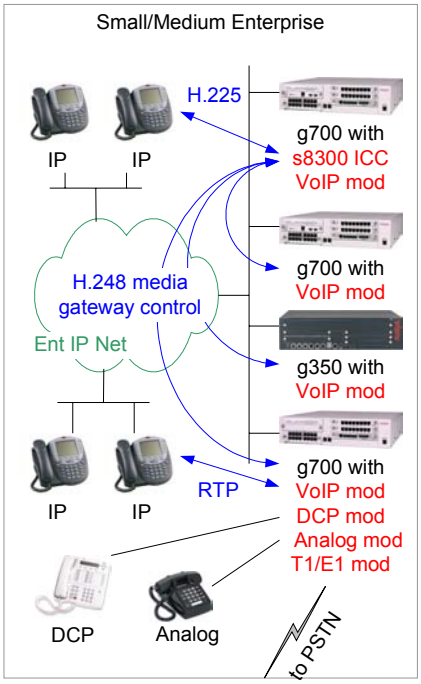


Figure 6: S8300/G700/G350/G250 architecture

Multi-Connect with Remote G700/G350/G250 Gateways

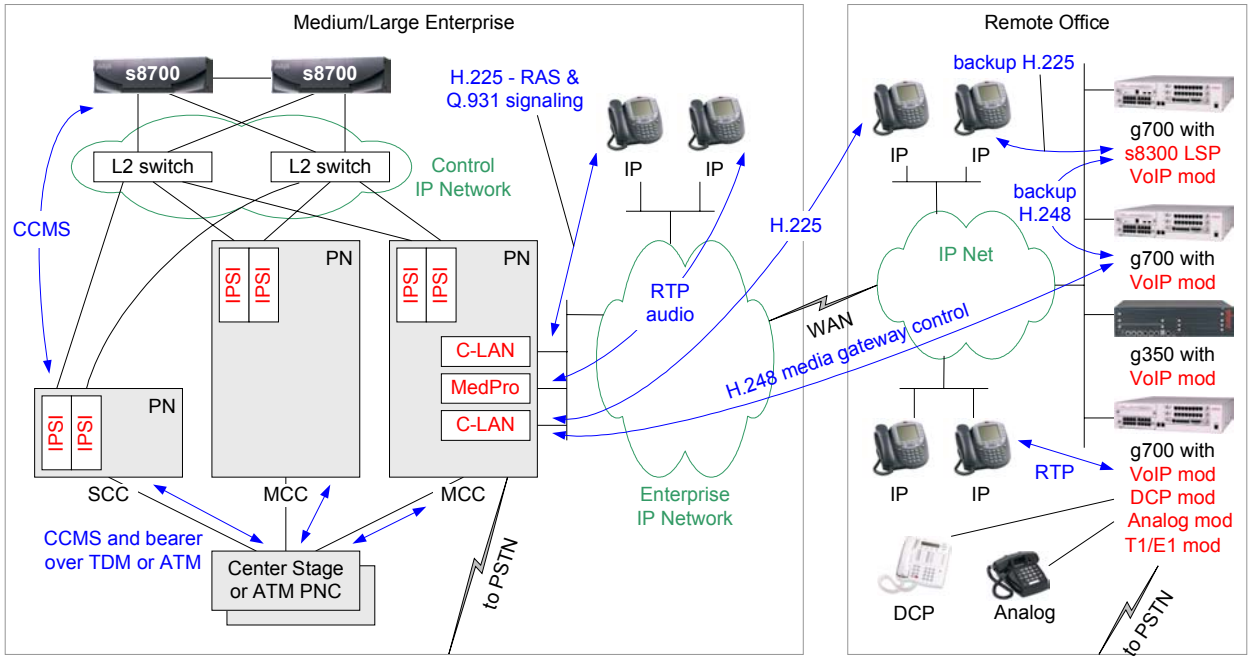


Figure 7: Multi-Connect with remote G700/G350/G250s

- Remote gateways and stations are controlled by the S87xx servers via the C-LAN boards.
- The remote S8300 is in local survivable processor (LSP) mode to take over as call server if connectivity to the S87xx servers is lost.

IP-Connect with Remote G700/G350/G250 Gateways

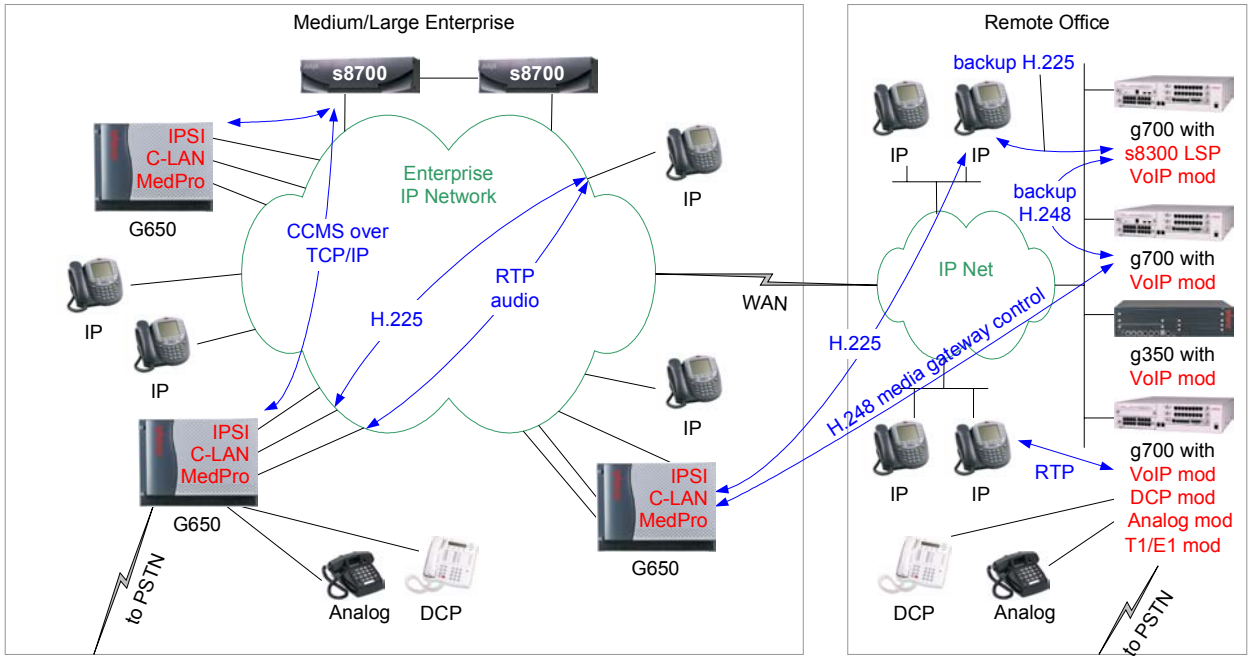


Figure 8: IP-Connect with remote G700/G350/G250s

- Remote gateways and stations are controlled by the S87xx servers via the C-LAN boards.
- The remote S8300 is in local survivable processor (LSP) mode to take over as call server if connectivity to the S87xx servers is lost.

Trunks

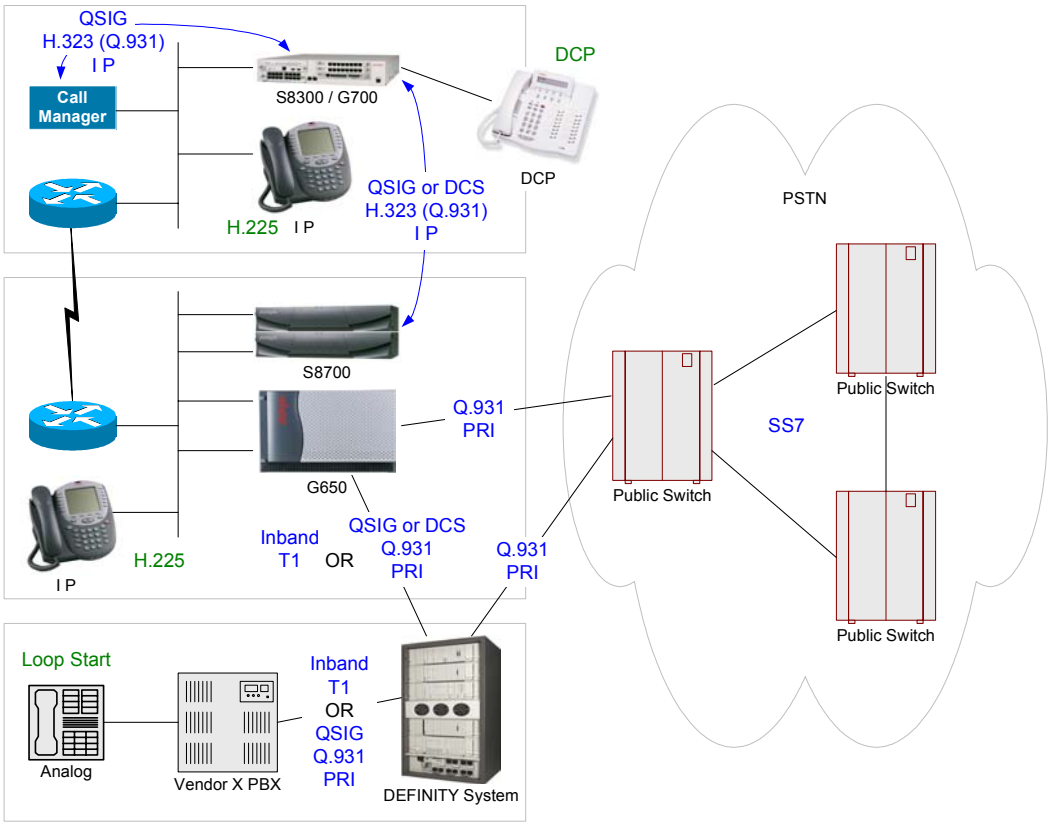


Figure 9: Trunks

This figure illustrates a broad picture to put trunks into context.

- PSTN trunks use the Signaling System 7 (SS7) signaling protocol. This protocol is not relevant to private, enterprise telephony systems.
- Private systems, such as the IP-Connect and DEFINITY servers in this illustration, commonly connect to public switches using ISDN PRI trunks with Q.931 signaling.
- Two private systems commonly connect to one another using T1 trunks with inband signaling, or ISDN PRI trunks with Q.931 signaling. This is illustrated in the trunks connecting the DEFINITY server to the IP-Connect, and to the Vendor X PBX.
- QSIG is a standard, feature-rich signaling protocol for private systems, and it "rides on top of" Q.931 as illustrated between the DEFINITY server and Vendor X PBX. DCS is the Avaya proprietary equivalent to QSIG, which also rides on top of Q.931 as illustrated between the IP-Connect and DEFINITY server.
- Gatekeepers, such as the S8700, S8300 and S8500, and Cisco Call Manager in this illustration, can connect to one another using IP trunks. The medium is IP and the signaling protocol is H.323, but Q.931 is still the fundamental component of H.323 that does the call signaling. And, as with ISDN PRI trunks, QSIG or DCS can be overlaid on top of Q.931.

QSIG is the standard signaling protocol that provides the feature-richness expected in enterprises. Generally speaking, traditional telephony systems support a broad range of QSIG features, while pure IP telephony systems support a very limited range. Due to the history and leadership of Avaya in traditional telephony, all Avaya call servers – whether traditional, IP-enabled, or pure IP – support virtually the same broad range of QSIG features.

1.3 VoIP Protocols and Ports

The following figure illustrates the protocol stacks relevant to VoIP. The contents of the upper-layer protocol messages are important to those who develop VoIP applications. However, those who implement these applications are typically not concerned with decoding the upper-layer messages. Instead, they are concerned primarily with the transport mechanism – TCP and UDP ports – so that they can verify and filter these message exchanges.

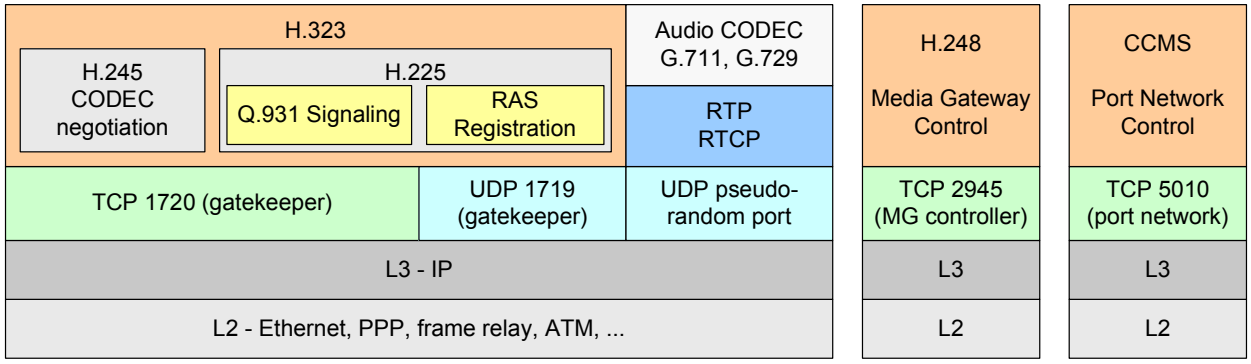


Figure 10: VoIP protocol stacks

- H.323 is the prevalent VoIP protocol suite. It is used for signaling from gatekeeper to terminals (stations), and gatekeeper to gatekeeper (trunks).
  - H.225 is the endpoint registration (RAS) and call signaling (Q.931) component of H.323.
    - H.225 call signaling messages are transported via TCP with port 1720 on the gatekeeper side.
    - H.225 registration messages (commonly referred to simply as RAS message) are sent via UDP with port 1719 on the gatekeeper side.
  - H.245 is the multimedia control component of H.323.
- Audio is digitally encoded prior to transmission and decoded after transmission using a coder/decoder (codec).
  - G.711 is the fundamental codec based on traditional pulse-code modulation (PCM), and it is generally recommended for LAN transport.
  - G.729 is a compressed codec, and it is generally recommended for transport over limited-bandwidth WAN links.
- Encoded audio is encapsulated in RTP (real-time protocol), then UDP, then IP.
  - RTP has fields such as Sequence Number and Timestamp that are designed for the transport and management of real-time applications.
  - On Avaya solutions the UDP ports used to transport RTP streams are configured on the call server.
  - Most protocol analyzers can identify RTP packets, making it easy to verify that audio streams are being sent between endpoints.
- H.248 is a protocol for media gateway control. It is transported via TCP with port 2945 (1039 if encrypted) on the media gateway controller side.
- CCMS is an Avaya proprietary protocol for port network control (same as media gateway control). It is transported via TCP with port 5010 on the port network (IPSI board) side.

2 IP Network Guidelines

This section gives general guidelines and addresses several issues related to IP networks (LAN/WAN) and device configurations.

2.1 General Guidelines

Because of the time-sensitive nature of VoIP applications, VoIP should be implemented on an entirely switched network. Ethernet collisions – a major contributor to delay and jitter – are virtually eliminated on switched networks. Additionally, VoIP endpoints should be placed on separate subnets or VLANs (separated from other non-VoIP hosts), with preferably no more than ~500 hosts per VLAN. This provides a cleaner design where VoIP hosts are not subjected to broadcasts from other hosts, and where troubleshooting is simplified. This also provides a routed boundary between the VoIP segments and the rest of the enterprise network, where restrictions can be placed to prevent unwanted traffic from crossing the boundary. When a PC is attached to an IP telephone, even if they are on separate VLANs, both PC and phone traffic (including broadcasts) traverse the same uplink to the Ethernet switch. In such a case the uplink should be a 100M link, and the recommended subnet/VLAN size is no larger than ~250 hosts each for the voice and data VLANs.

High broadcast levels are particularly disruptive to real-time applications like VoIP. Avaya media servers and gateways and IP telephones utilize very low amounts of broadcast traffic to operate. Therefore, a subnet/VLAN with only these Avaya hosts has a very low broadcast level. There are two cases where Avaya hosts can be subjected to high levels of broadcasts: 1) Avaya hosts and other broadcast-intensive hosts share a subnet/VLAN; and 2) broadcast-intensive PCs are attached to Avaya IP phones. Case 1 is one of the reasons for the recommendation to use separate voice subnets/VLANs. Case 2 is sometimes unavoidable, and the result is that broadcasts used by the PC must pass through the phone, even if the phone and PC are on different VLANs. For this reason Avaya IP phones are designed to be very resilient against broadcasts, with lab tests showing the phones operating satisfactorily even with 3,000 to 10,000 broadcasts per second, depending on the model. Nevertheless, to provide acceptable user experience and audio quality, high-broadcast environments are very strongly discouraged. The recommended maximum broadcast rate is 500 per second, and the absolute maximum is 1000 per second.

If VoIP hosts must share a subnet with non-VoIP hosts (not recommended), they should be placed on a subnet/VLAN of ~250 hosts or less with as low a broadcast rate as possible. Use 100M links, take measures not to exceed the recommended maximum broadcast rate (500/s), and do not exceed the absolute maximum broadcast rate (1000/s).

In summary, the general guidelines are…

- All switched network; no hubs.
- Separate voice VLANs.
- No more than ~500 hosts (/23 subnet mask) on voice VLAN with only VoIP endpoints.
- No more than ~250 hosts (/24 subnet mask) each on voice and data VLANs if IP phones have PCs attached to them.
- 100M uplink between IP phone and Ethernet switch.
- As low a broadcast rate as possible – 500/s recommended max; 1000/s absolute max.

Ethernet Switches

The following recommendations apply to Ethernet switches to optimize operation with Avaya IP telephones and other Avaya VoIP endpoints, such as IP boards. They are meant to provide the simplest configuration by removing unnecessary features.

- Enable Spanning Tree fast start feature or disable Spanning Tree on host ports – The Spanning Tree Protocol (STP) is a layer 2 (L2) protocol used to prevent loops when multiple L2 network devices are connected together. When a device is first connected (or re-connected) to a port running STP, the port takes approximately 50 seconds to cycle through the Listening, Learning, and Forwarding states.

This 50-second delay is not necessary and not desired on ports connected to IP hosts (non-network devices). Enable a fast start feature on these ports to put them into the Forwarding state almost immediately. Avaya P550 calls this **fast-start** and Cisco calls it **portfast**. If this feature is not available, disabling STP on the port is an option that should be considered. *Do not disable STP on an entire switch or VLAN.*

- Enable Rapid Spanning Tree and configure host ports as edge ports – As the name implies, Rapid Spanning Tree Protocol (RSTP) is a faster and more advanced replacement for STP. RSTP is preferred over STP when all network devices in a L2 domain support it. Even if they don't, there are ways to combine RSTP and STP (depending on the network equipment), though certainly not as clean as having RSTP throughout the L2 domain. When running RSTP, configure the host ports as **edge** ports, which is equivalent to enabling fast-start or portfast in a STP domain.
- Disable Cisco features – Cisco features that are not required by Avaya endpoints are **auxiliaryvlan** (except for IP phones in a dual-VLAN setting as described in appendices A and B), **channeling**, **cdp**, **inlinepower**, and any Cisco proprietary feature in general. Explicitly disable these features on ports connected to Avaya devices, as they are non-standard mechanisms relevant only to Cisco devices and can sometimes interfere with Avaya devices. The CatOS command **set port host <mod/port>** automatically disables channeling and trunking, and enables portfast. Execute this command first, and then manually disable cdp, inlinepower, and auxiliaryvlan. *For dual-VLAN IP telephone implementations, see Appendices A and B for more information and updates regarding auxiliaryvlan and trunking.*
- Properly configure 802.1Q trunking on Cisco switches – If trunking is required on a Cisco CatOS switch connected to an Avaya device, enable it for 802.1Q encapsulation in the *nonegotiate* mode (**set trunk <mod/port> nonegotiate dot1q**). This causes the port to become a plain 802.1Q trunk port with no Cisco auto-negotiation features. When trunking is not required, explicitly disable it, as the default is to auto-negotiate trunking.

Speed/Duplex

One major issue with Ethernet connectivity is proper configuration of speed and duplex. There is a significant amount of misunderstanding in the industry as a whole regarding the auto-negotiation standard. The speed can be sensed, but the duplex setting is negotiated. This means that if a device with fixed speed and duplex is connected to a device in auto-negotiation mode, the auto-negotiating device can sense the other device's speed and match it. But the auto-negotiating device cannot sense the other device's duplex setting; the duplex setting is negotiated. Therefore, the auto-negotiating device always goes to half duplex in this scenario. The following table is provided as a quick reference for how speed and duplex settings are determined and typically configured. It is imperative that the speed and duplex settings be configured properly.

Device1 Configuration	Device2 Configuration	Result
auto-negotiate	auto-negotiate	100/full expected and often achieved, but not always stable. Suitable for user PC connections, but not suitable for server connections or uplinks between network devices. Suitable for a single VoIP call, such as with a softphone or single IP telephone. Not suitable for multiple VoIP calls, such as through a MedPro/MR320 board.
auto-negotiate	100/half	100/half stable. Device1 senses the speed and matches accordingly. Device1 senses no duplex negotiation, so it goes to half duplex.
auto-negotiate	10/half	10/half stable. Device1 senses the speed and matches accordingly. Device1 senses no duplex negotiation, so it goes to half duplex.
auto-negotiate	100/full	Device1 goes to 100/half, resulting in a duplex mismatch – undesirable. Device1 senses the speed and matches accordingly. Device1 senses no duplex negotiation, so it goes to half duplex.

100/full	100/full	100/full stable. Typical configuration for server connections and uplinks between network devices.
10/half 100/half	10/half 100/half	Stable at respective speed and duplex. Some enterprises do this on user ports as a matter of policy for various reasons.

Table 1: Speed/duplex matrix

Layer 1 (L1) errors such as *runts*, *CRC errors*, *FCS errors*, and *alignment errors* often accompany a duplex mismatch. If these errors exist and continue to increment, there is probably a duplex mismatch or cabling problem or some other physical layer problem. The **show port <mod/port>** command on Catalyst switches gives this information. The Avaya P550 commands are **show port status <mod/port>**, **show port counters <mod/port>**, and **show ethernet counters <mod/port>**. The Avaya P330 switch command is **show rmon statistics <mod/port>**.

2.2 Bandwidth Considerations

Calculation

Many VoIP bandwidth calculation tools are available, as well as pre-calculated tables. Rather than presenting a table, the following information is provided to help the administrator make an informed bandwidth calculation. The per-call rates for G.711, G.726 and G.729 are provided under the "Ethernet Overhead" and "WAN Overhead" headings below, and all calculations are for the recommended voice packet size of 20ms.

- Voice payload and codec selection – The G.711 codec payload rate is 64000bps. Since the audio is encapsulated in 10-ms frames, and there are 100 of these frames in a second (100 * 10ms = 1s), each frame contains 640 bits (64000 / 100) or 80 bytes of voice payload. The G.726 codec payload rate is 32000bps and the G.729 codec payload rate is 8000bps. This equates to 320 bits or 40 bytes and 80 bits or 10 bytes per 10-ms frame respectively

Voice Payload	1 frame – 10ms	2 frames – 20ms	3 frames – 30ms	4 frames – 40ms
G.711	80 B	160 B	240 B	320 B
G.726	40 B	80 B	120 B	160 B
G.729	10 B	20 B	30 B	40 B

Table 2: Voice payload vs. number of frames

- Packet size and packet rate – Because the voice payload rate must remain constant, the number of voice frames per packet (packet size) determines the packet rate. As the number of frames per packet increases, the number of packets per second decreases to maintain a steady rate of 100 voice frames per second.

Packet Rate	Codec Payload Rate	1 frame/packet 10ms	2 frames/packet 20ms	3 frames/packet 30ms	4 frames/packet 40ms
G.711	64000bps	100pps	50pps	33pps	25pps
G.726	32000bps	100pps	50pps	33pps	25pps
G.729	8000bps	100pps	50pps	33pps	25pps

Table 3: Packet rate vs. packet size

- IP, UDP, RTP overhead – Each voice packet inherits a fixed overhead of 40 bytes.

IP 20 B	UDP 8 B	RTP 12 B	Voice Payload Variable

Figure 11: IP/UDP/RTP overhead

To this point the calculation is simple. Add up the voice payload and overhead per packet, and multiply by the number of packets per second. Here are the calculations for a G.711 and a G.729 call, both using 20-ms packets. (Remember that there are 8 bits per byte.)

G.711: (160B voice payload + 40B overhead)/packet * 8b/B * 50 packets/s = 80kbps
G.726: (80B Voice payload + 40B overhead/packet *8b/B * 50 packets/s = 48kbps
G.729: (20B voice payload + 40B overhead)/packet * 8b/B * 50 packets/s = 24kbps

The calculations above do not include the L2 encapsulation overhead. L2 overhead must be added to the bandwidth calculation, and this varies with the protocol being used (Ethernet, PPP, HDLC, ATM, Frame Relay, etc).

L2 header	IP 20 B	UDP 8 B	RTP 12 B	Voice Payload Variable	L2 trailer

Figure 12: L2 overhead

Ethernet Overhead

G.711 20-ms call over Ethernet = 90.4kbps
G.711 30-ms call over Ethernet = 81.6kbps
G.726 20-ms call over Ethernet = 58.4kbps
G.726 30-ms call over Ethernet = 49.1kbps
G.729 20-ms call over Ethernet = 34.4kbps
G.729 30-ms call over Ethernet = 25.6kbps

Ethernet has a header of 14 bytes and a trailer of 4 bytes. It also has a 7-byte preamble and a 1-byte start of frame delimiter (SFD), which some bandwidth calculation tools do not take into consideration. Nevertheless, the preamble and SFD consume bandwidth on the LAN, so the total Ethernet overhead is 26 bytes. When transmitting 20-ms voice packets, the Ethernet overhead equates to 10.4kbps (26 * 8 * 50), which must be added to the 80kbps for G.711, 40kbps for G.726, and 24kbps for G.729. For full-duplex operation the totals are 90.4kbps for G.711, 50.4kbps for G.726, and 34.4kbps for G.729. For half-duplex operation these figures are at least doubled, but effectively the load is higher due to the added overhead of collisions.

WAN Overhead

The WAN overhead is calculated just like the Ethernet overhead, by determining the size of the L2 encapsulation and figuring it into the calculation. L2 headers and trailers vary in size with the protocol being used, but they are typically much smaller than the Ethernet header and trailer. For example, the PPP overhead is only 7 bytes. However, to allow for a high margin of error, assume a 14-byte total L2 encapsulation size, which would add an overhead of 5.6kbps (14 * 8 * 50), assuming 20-ms voice packets. This would result in approximately 43kbps for G.726 and 30kbps for G.729 over a WAN link. Significant bandwidth savings are realized by using a compressed codec (G.729 recommended) across a WAN link. *Note that in today's data networks most, if not all, WAN links are full duplex.*

G.729 20-ms call over PPP = 26.8kbps
G.726 20-ms call over PPP = 50.8kbps

G.729 20-ms call over 14-B L2 = 29.6kbps
G.726 20-ms call over 14-B L2 = 53.6kbps

L3 Fragmentation (MTU)

Related to bandwidth, there are two other factors that must be considered for operation across WAN links, and they both involve fragmentation. The first factor, maximum transmission unit (MTU), involves fragmenting the layer 3 (L3) payload. The MTU is the total size of the L3 packet (IP header + IP payload), which is 200 bytes for G.711 and 60 bytes for G.729 (assuming 20-ms voice packets). If the MTU on an interface is set below these values the IP payload (UDP + RTP + voice payload) must be fragmented into multiple IP packets, each packet incurring the 20-byte IP overhead. For example,

suppose the MTU on an interface is set to 100 bytes, which is an extremely low value. The 20-ms G.711 IP packet is 200 bytes, consisting of a 20-byte IP header and a 180-byte IP payload. The 180-byte payload must be divided into three fragments of 80 bytes, 80 bytes, and 20 bytes. Each fragment incurs a 20-byte IP header to make the packets 100 bytes, 100 bytes, and 40 bytes. A single 200-byte IP packet must be fragmented into three separate IP packets to meet the 100-byte MTU. In addition, the L2 overhead also increases because each L3 packet requires L2 encapsulation.
MTU should not be an issue for VoIP because most interfaces have a default MTU of 1500 bytes. However, it is possible to intentionally set the MTU to low levels. Even if the MTU is not set to a level that would fragment VoIP packets, the principle of fragmenting the L3 payload and incurring additional L3 and L2 overhead applies universally. Changing the MTU requires a thorough understanding of the traffic traversing the network. A low MTU value, relative to any given IP packet size, always increases L3 and L2 overhead as illustrated with the VoIP example. Because of this inefficiency, it is generally not advisable to lower the MTU.

L2 Fragmentation

The second factor involves fragmenting the L2 payload, or the entire IP packet. This process of fragmenting a single IP packet into multiple L2 frames incurs additional L2 overhead, but no additional IP overhead. For example, the fixed cell size for ATM is 53 octets (bytes), with 5 octets for ATM overhead and 48 octets for payload. Without header compression there is no way to get a voice packet to fit inside one ATM cell. Therefore, the L3 packet (not just the IP payload, but the entire IP packet) is fragmented and carried inside multiple ATM cells. A 200-byte G.711 IP packet would require five ATM cells (25 octets of ATM overhead), whereas a 60-byte G.729 IP packet would only require two ATM cells (10 octets of ATM overhead). *Refer to Appendix C for information regarding RTP header compression. Keep in mind, however, that the same precautions apply to RTP header compression as to QoS (see the next section on CoS and QoS). The router could pay a significant processor penalty if the compression is done in software.*
Inter-LATA (typically interstate) Frame Relay is also affected by this ATM phenomenon. This is because most carriers (ATT, Verizon, Sprint) convert Frame Relay to ATM for the long haul, between the local central offices. This is done through a process called frame-relay-to-ATM network interworking and service interworking (FRF.5 and FRF.8). In this process the Frame Relay header is translated to an ATM header, and the Frame Relay payload is transferred to an ATM cell. Since the Frame Relay payload can be a variable size but the ATM payload is a fixed size, a single Frame Relay frame can be converted to multiple ATM cells for the long haul. Therefore, it is beneficial to limit the size of the voice packet even when the WAN link is Frame Relay.

2.3 CoS and QoS

General

The term “Class of Service” refers to mechanisms that mark traffic in such a way that the traffic can be differentiated and segregated into various classes. The term “Quality of Service” refers to what the network does to the marked traffic to give higher priority to specific classes. If an endpoint marks its traffic with L2 802.1p priority 6 and L3 DSCP 46, for example, the Ethernet switch must be configured to give priority to value 6, and the router must be configured to give priority to DSCP 46. The fact that certain traffic is marked with the intent to give it higher priority does not necessarily mean it receives higher priority. CoS marking does no good without the supporting QoS mechanisms in the network devices.

CoS

802.1p/Q at the Ethernet layer (L2) and DSCP at the IP layer (L3) are two CoS mechanisms that Avaya products employ. These mechanisms are supported by the IP telephones and most IP port boards. In addition, the call server can flexibly assign the UDP port range for audio traffic transmitted from the MedPro/MR320 board or VoIP media module. Although TCP/UDP source and destination ports are not

CoS mechanisms, they are inherently used to identify specific traffic and can be used much like CoS markings. Other non-CoS methods to identify specific traffic are to key in on source and destination IP addresses and specific protocols (ie, RTP).

802.1p/Q

The figure below shows the IEEE 802.1Q tag and its insertion point in the Ethernet and 802.3 frames. Note that in both cases the 802.1Q tag changes the size and format of the comprehensive Ethernet and 802.3 frames. Because of this, many intelligent switches (ones that examine the L2 header and perform some kind of check against the L2 frame) must be explicitly configured to accept 802.1Q tagged frames. Otherwise, these switches may reject the tagged frames. The Tag Protocol Identifier (TPID) field has hex value x8100 (802.1QTagType). This value alerts the switch or host that this is a tagged frame. If the switch or host does not understand 802.1Q tagging, the TPID field is mistaken for the Type or Length field, which results in an erroneous condition.

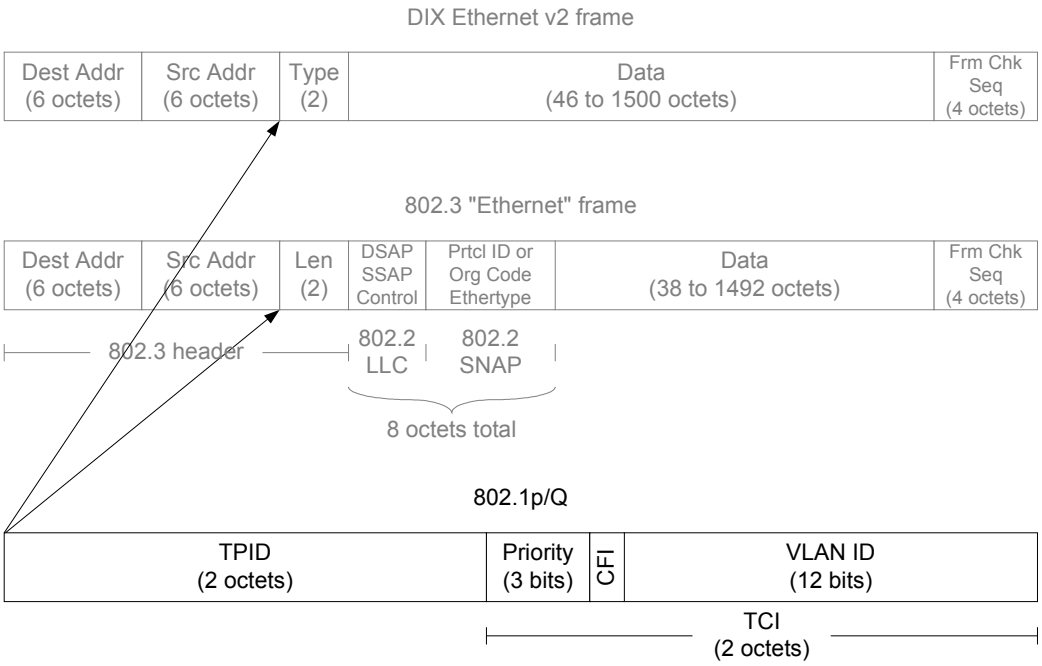


Figure 13: 802.1Q tag

The two other fields of importance are the Priority and Vlan ID (VID) fields. The Priority field is the "p" in 802.1p/Q and ranges in value from 0 to 7. ("802.1p/Q" is a common term used to indicate that the Priority field in the 802.1Q tag has significance. Prior to real-time applications 802.1Q was used primarily for VLAN trunking, and the Priority field was not important.) The VID field is used as it always has been – to indicate the VLAN to which the Ethernet frame belongs.

Rules for 802.1p/Q Tagging

There are two questions that determine when and how to tag:

1. Is tagging required to place the frame on a specific VLAN (**VLAN tagging**)?
2. Is tagging required to give the frame a priority level greater than 0 (**priority tagging**)?

Based on the answers to these questions, tagging should be enabled following these two rules.

1. Single-VLAN Ethernet switch port (default scenario).
   - On a single-VLAN port there is no need to tag to specify a VLAN, because there is only one VLAN.
   - For priority tagging only, the IEEE 802.1Q standard specifies the use of VID 0. VID 0 means that the frame belongs on the port's primary VLAN, which IEEE calls the "port VLAN," and

Cisco calls the "native VLAN." Some Ethernet switches do not properly interpret VID 0, in which case the port/native VID may need to be used, but this is not the standard method.
- For single devices, such as a call server or port board, a simpler alternative is to not tag at all, but configure the Ethernet switch port as a high-priority port instead. This treats all incoming traffic on that port as high-priority traffic, based on the configured level.
- For multiple devices on the same VLAN, such as an IP telephone with a PC attached, the high-priority device (IP telephone) should tag with VID 0 and the desired priority. The low-priority device (PC) would not tag at all. No tag at all is the same as priority 0 (default).

2. Multi-VLAN Ethernet switch port.
   - A multi-VLAN port has a single port/native VLAN and one or more additional tagged VLANs, with each VLAN pertaining to a different IP subnet.
   - In general, do not configure multiple VLANs on a port with only one device attached to it (unless that device is another Ethernet switch across a trunk link).
   - For the attached device that belongs on the port/native VLAN, follow the points given for rule 1 above. Clear frames (untagged frames) are forwarded on the port/native VLAN by default.
   - An attached device that belongs on any of the tagged VLANs must tag with that VID and the desired priority.
   - The most common VoIP scenario for a multi-VLAN port is an IP telephone with a PC attached, where the phone and PC are on different VLANs. In this case the PC would send clear frames, and the IP telephone should tag with the designated VID and desired priority.

As stated previously, an Ethernet switch must be capable of interpreting the 802.1Q tag, and many must be explicitly configured to receive it. The use of VID 0 is a special case, because it only specifies a priority and not a VLAN. Avaya switches accept VID 0 without any special configuration, but some Ethernet switches do not properly interpret VID 0. And some switches require trunking to be enabled to accept VID 0, while others do not. The following table shows the results of some testing performed by Avaya Labs on Cisco switches.

Catalyst 6509 w/ CatOS 6.1(2)	Accepted VID 0 for the native VLAN when 802.1Q trunking was enabled on the port.
Catalyst 4000 w/ CatOS 6.3(3)	Would not accept VID 0 for the native VLAN. Opened a case with Cisco TAC, and TAC engineer said it was a hardware problem in the 4000. Bug ID is CSCdr06231. Workaround is to enable 802.1Q trunking and tag with native VID instead of 0.
Catalyst 3500XL w/ IOS 12.0(5)WC2	Accepted VID 0 for the native VLAN when 802.1Q trunking was disabled on the port.
Conclusion	Note the hardware platform and OS version and consult Cisco's documentation, or call TAC.

Table 4: Sample VID 0 behaviors for Cisco switches

See Appendix A for more information on VLANs and tagging.

DSCP

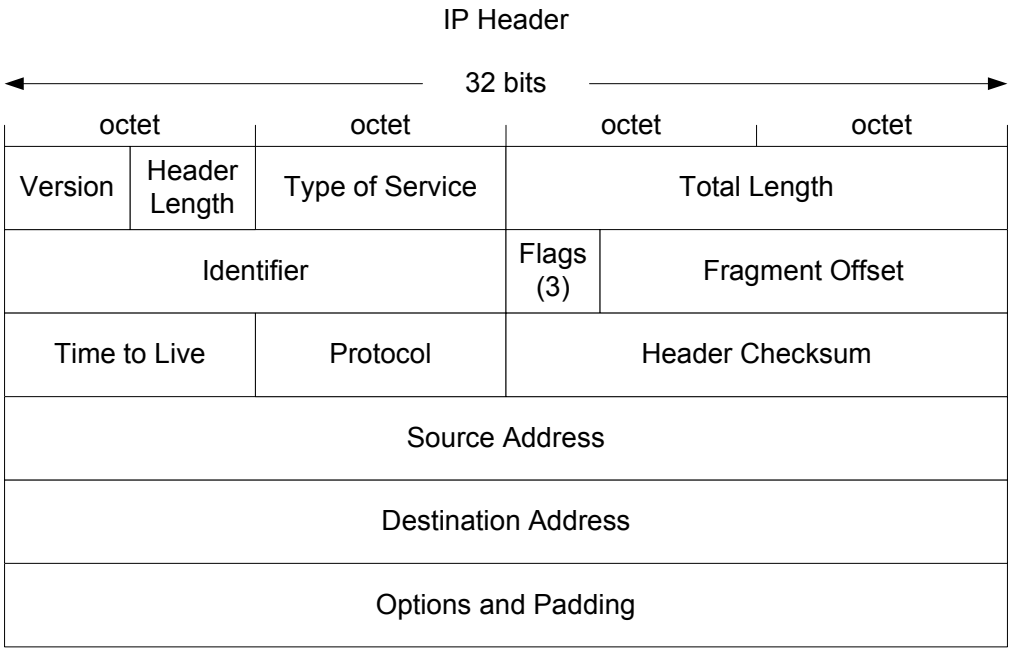


Figure 14: IP header

The figure above shows the IP header with its 8-bit Type of Service (ToS) field. The ToS field contains three IP Precedence bits and four Type of Service bits as follows.

Bits 0-2 IP Precedence	000 001 010 011 100 101 110 111	Routine Priority Immediate Flash Flash Override CRITIC/ECP Internetwork Control Network Control
Bit 3 Delay	0 1	Normal Low
Bit 4 Throughput	0 1	Normal High
Bit 5 Reliability	0 1	Normal High
Bit 6 Monetary Cost	0 1	Normal Low
Bit 7 Reserved		Always set to 0

Figure 15: Original scheme for IP ToS field

This original scheme was not widely used, and the IETF came up with a new marking method for IP called Differentiated Services Code Points (DSCP, RFC 2474/2475). DSCP utilizes the first six bits of the ToS field and ranges in value from 0 to 63. The following figure shows the original ToS scheme and DSCP in relation to the eight bits of the ToS field.

8-bit Type of Service field							
IP Precedence bits			Type of Service bits				0
0	1	2	3	4	5	6	7
DSCP bits						0	0

Figure 16: Compare DSCP w/ original ToS

Ideally any DSCP value would map directly to a Precedence/ToS combination of the original scheme. This is not always the case, however, and it can cause problems on some legacy devices, as explained in the following paragraph.

On any device, new or old, having a non-zero value in the ToS field cannot hurt if the device is not configured to examine the ToS field. The problems arise on some legacy devices when the ToS field is examined, either by default or by enabling QoS. These legacy devices (network and endpoint) may contain code that only implemented the IP Precedence portion of the original ToS scheme, with the remaining bits defaulted to zeros. This means that only DSCP values divisible by 8 (XXX000) can map to the original ToS scheme. For example, if an endpoint is marking with DSCP 40, a legacy network device can be configured to look for IP Precedence 5, because both values show up as 10100000 in the ToS field. However, a DSCP of 46 (101110) cannot be mapped to any IP Precedence value alone. Another hurdle is if the legacy code implemented IP Precedence with only one ToS bit permitted to be set high. In this case a DSCP of 46 still would not work, because it would require two ToS bits to be set high. When these mismatches occur, the legacy device may reject the DSCP-marked IP packet or exhibit some other abnormal behavior. Most newer devices support both DSCP and the original ToS scheme.

QoS on an Ethernet Switch

On Avaya and Cisco Catalyst switches, VoIP traffic can be assigned to higher priority queues. Queuing is very device-dependent, but in general an Ethernet switch has a fixed number of queues that may be configurable, but are typically defaulted to optimized settings for most implementations. The number of queues and the technical sophistication of the queuing vary among switches, but in general the more advanced the switch, the more granular the queuing to service the eight L2 priority levels. An Ethernet switch can classify traffic based on the 802.1p/Q priority tag and assign each class of traffic to a specific queue, but only if this is a default feature or it is explicitly configured. On many switches a specific port can be designated as a high priority port, causing all incoming traffic on that port to be assigned to a high priority queue. This frees the endpoint from having to tag its traffic with L2 priority.

QoS on a Router

It is generally more complicated to implement QoS on a router than on an Ethernet switch. Unlike Ethernet switches, routers typically do not have a fixed number of queues. Instead, routers have various queuing mechanisms. For example, Cisco routers have standard first-in first-out queuing (FIFO), weighted fair queuing (WFQ), custom queuing (CQ), priority queuing (PQ), and low latency queuing (LLQ). LLQ is a combination of priority queuing and class-based weighted fair queuing (CBWFQ), and it is Cisco's recommended queuing mechanism for real-time applications such as VoIP. Each queuing mechanism behaves differently and is configured differently, but following a common sequence. First the desired traffic must be classified using DSCP, IP address, TCP/UDP port, or protocol. Then the traffic must be assigned to a queue in one of the queuing mechanisms. Then the queuing mechanism must be applied to an interface. [2 p.1-7, 3-4, 3-5, 5-2]

The interface itself may also require additional modifications, independent of the queuing mechanism, to make QoS work properly. For example, Cisco requires traffic shaping on Frame Relay and ATM links to help ensure that voice traffic is allotted the committed or guaranteed bandwidth (see "Traffic Shaping on Frame Relay Links" below in this section). Cisco also recommends link fragmentation and interleaving (LFI) on WAN links below 768kbps, to reduce serialization delay. Serialization delay is the delay incurred in encapsulating a L3 packet in a L2 frame and transmitting it out the serial interface. It increases with packet size but decreases with WAN link size. The concern is that large, low priority packets induce additional delay and jitter, even with QoS enabled. This is overcome by fragmenting the large, low priority packets and interleaving them with the small, high priority packets, thus reducing the wait time for the high priority packets. The following matrix is taken directly from the "Cisco IP Telephony QoS Design Guide" [2 p.1-3].

WAN Link Speed	L3 Packet Size					
	64 bytes	128 bytes	256 bytes	512 bytes	1024 bytes	1500 bytes
56 kbps	9 ms	18 ms	36 ms	72 ms	144 ms	214 ms
64 kbps	8 ms	16 ms	32 ms	64 ms	128 ms	187 ms
128 kbps	4 ms	8 ms	16 ms	32 ms	64 ms	93 ms
256 kbps	2 ms	4 ms	8 ms	16 ms	32 ms	46 ms
512 kbps	1 ms	2 ms	4 ms	8 ms	16 ms	23 ms
768 kbps	640 *u*s	1.28 ms	2.56 ms	5.12 ms	10.24 ms	15 ms

Table 5: Cisco seralization delay matrix

Consult Cisco's documentation for detailed information regarding traffic shaping and LFI, and be especially careful with LFI. On one hand it reduces the serialization delay, but on the other it increases the amount of L2 overhead. This is because a single L3 packet that was once transported in a single L2 frame, is now fragmented and transported in multiple L2 frames. Configure the fragment size to be as large as possible while still allowing for acceptable voice quality.

Instead of implementing LFI, some choose to simply lower the MTU size to reduce serialization delay. Two possible reasons for this are that LFI may not be supported on a given interface, or that lowering the MTU is easier to configure. As explained in section 2.2 under the heading "L3 Fragmentation (MTU)," lowering the MTU (L3 fragmentation) is much less efficient than LFI (L2 fragmentation) because it incurs additional L3 overhead as well as additional L2 overhead. Lowering the MTU is generally not advisable and may not provide any added value, because it adds more traffic to the WAN link than LFI. The added congestion resulting from the increase in traffic may effectively negate any benefit gained from reducing serialization delay. One should have a thorough understanding of the traffic traversing the WAN link before changing the MTU.

Because of all these configuration variables, properly implementing QoS on a router is no trivial task. However, it is on the router where QoS is needed most, because most WAN circuits terminate on routers. Appendix F contains examples of implementing QoS on Cisco routers. This appendix does not contain configurations for all the issues discussed in this document, but it gives the reader a place to start.

QoS Guidelines

There is no all-inclusive set of rules regarding the implementation of QoS, because all networks and their traffic characteristics are unique. It is good practice to baseline the VoIP response (ie, voice quality) on a network without QoS, and then apply QoS as necessary. Conversely, it is very bad practice to enable multiple QoS features simultaneously, not knowing what effects, if any, each feature is introducing. If voice quality is acceptable without QoS, then the simplest design may be a wise choice. If voice quality is not acceptable, or if QoS is desired for contingencies such as unexpected traffic storms, the best place to begin implementing QoS is on the WAN link(s). Then QoS can be implemented on the LAN segments as necessary.

One caution to keep in mind about QoS is regarding the processor load on network devices. Simple routing and switching technologies have been around for many years and have advanced significantly. Packet forwarding at L2 and L3 is commonly done in hardware (Cisco calls this *fast switching* [2 p.5-18], "switching" being used as a generic term here), without heavy processor intervention. When selection criteria such as QoS and other policies are added to the routing and switching function, it inherently requires more processing resources from the network device. Many of the newer devices can handle this additional processing in hardware, resulting in maintained speed without a significant processor burden. However, to implement QoS, some devices must take a hardware function and move it to software (Cisco calls this *process switching* [2 p.5-18]). Process switching not only reduces the speed of packet forwarding, but it also adds a processor penalty that can be significant. This can result in an overall performance degradation from the network device, and even device failure.

Each network device must be examined individually to determine if enabling QoS will reduce its overall effectiveness by moving a hardware function to software, or for any other reason. Since most QoS

policies are implemented on WAN links, the following very general points for Cisco routers are offered to increase the level of confidence that QoS remains in hardware. **Consult Cisco to be sure.**

- Newer hardware platforms are required: 2600, 3600, 7200, and 7500.
- Newer interface modules (WIC, VIP, etc.) are required: Consult Cisco to determine which hardware revision is required for any given module.
- Sufficient memory is required: Device dependent.
- Newer IOS is required: 12.0 or later.

Several things should be examined whenever QoS is enabled on a network device. First, the processor level on the device should be examined and compared to levels before QoS was enabled. It is likely that the level will have gone up, but the increase should not be significant. If it is significant, then it is likely that the QoS process is being done by software. The processor load must remain at a manageable level (max 50% average, 80% peak). If the processor load is manageable, the VoIP response should be examined to verify that it has improved under stressed conditions (ie, high congestion) compared to performance before QoS was implemented. There is no added value in leaving a particular QoS mechanism enabled if VoIP response has not improved under stressed conditions. If VoIP response has improved, then the other applications should be checked to verify that their performances have not degraded to unacceptable levels.

Traffic Shaping on Frame Relay Links

Experience to date supports Cisco's requirement to use traffic shaping on frame relay links [2 p.5-22]. Simply stated, VoIP traffic must be sent within the committed information rate (CIR) and not in the burst range. This means that everything traversing a specific interface or sub-interface must be sent within CIR, because there is no mechanism to dictate that VoIP be sent within CIR while data is sent in the burst range on the same interface. Under this constraint one solution for maximizing bandwidth is to make the CIR as large as possible, and this is dictated by the end of the PVC that has the smaller access circuit. Consult each router vendor's documentation to see if other methods are available.

3 Guidelines for Avaya Servers and Gateways

This section gives guidelines for Avaya servers and gateways, and covers most of the IP-telephony-related configurations. Refer back to section 1 for an overview of IP telephony components and Avaya architectures.
Avaya Communication Manager is the call processing software that runs on Avaya servers, and it is configured via the Switch Administration Terminal (SAT) interface. Although the server platforms themselves may be configured in various ways, SAT is the universal interface for Communication Manager.
The Avaya Site Administrator (SA) is a client software application used to access the SAT interface on all Avaya servers. Additionally, on all but the DEFINITY servers, SAT can also be accessed by telnet-ing to the server.

3.1 S87xx/S8500 Servers

The S87xx and S8500 are 19-inch rack-mountable Red Hat Linux server platforms. S87xx servers operate in a redundant pair, whereas the S8500 is a simplex server. Each server is configured and managed via a variety of web interfaces, with the **Maintenance Web Interface** being the most comprehensive. The web interfaces are designed to facilitate all anticipated configuration and management requirements, and there is little or no need for a customer to access the Linux shell.
In an S87xx pair one of the servers is active and the other is standby. SAT administration is performed on the active server, and it is automatically carried over to the standby server. Either of the servers could be active or standby at any given time, and there are different ways to determine which is active. If the two servers are on the same subnet there is a virtual IP address, which is labeled the **active server** address in the **Configure Server – Configure Interfaces** screen of the Maintenance Web Interface. Whichever server is active takes control of the active server address, and telnet-ing or browsing or pointing Avaya SA to that address accesses the active server. If the two S8700 servers are not on the same subnet (server separation), there is no virtual active server address. The **Status Summary** web screen shows the status of the servers.

The S8700 SAT interface may be accessed using Avaya Site Administration (ASA) or by telnet-ing to port 5023: **telnet <active server address> 5023**. A SAT session can also be established by telnet-ing to the active server and typing **sat** from the Linux shell. The standby server does not permit access to SAT. Secure Shell (SSH) access is recommended for encrypted connections to an S87XX server pair. ASA supports Secure Shell (SSH) access for system administration. A SSH Client can also be used to access the SAT on port 5022.

SAT access to the S8500 is similar to that of the S87xx server pair, except that there is only one server.

As of CM 3.1 a S8500 main server or S8500 LSP supports Processor Ethernet (PE). The S8500 PE provides similar functions as a CLAN (TN799DP) Circuit Pack for H.323 IP endpoints, H.248 gateways and subset of adjuncts. An S8500 PE interface uses one of the native NICs on the server and allows for direct connections to H.248 Media Gateways without the need for port networks (CLAN + IPSI). There are, however, configuration limitations, which are defined in the Overview for AVAYA Communication Manager, Document ID 03-300468, available on the support.avaya.com website.

S87xx/S8500 Speed/Duplex

Speed and duplex for the various S87xx/S8500 Ethernet interfaces are configured using the **Configure Server – Configure Interfaces** web admin screen. It is critical to configure the speed and duplex correctly on the server interfaces used to communicate with the IPSI boards. A speed/duplex mismatch between these interfaces and the Ethernet switch causes severe call processing problems.

The web admin screen has a pull-down menu for the various speed/duplex settings. This pull-down menu does not indicate the current configuration, but only the available options. A "current speed" description next to this pull-down menu indicates the current speed and duplex, but it does not indicate whether these settings were manually configured or reached via auto-negotiation. Follow these steps to properly configure the speed and duplex.

- Start with the server and the Ethernet switch port set to auto-negotiate (default). The server should show "(Current speed: 100 Megabit full duplex)" on the web admin screen, and the Ethernet switch port should show that the negotiated speed/duplex is 100/full. When in doubt always return to this base state.
- On the server manually configure the interface to 100/full. With the Ethernet switch port still at auto-negotiate, it should now show that the negotiated speed/duplex is 100/half. This is expected.
- Manually configure the Ethernet switch port to 100/full. After a screen refresh the server should still show the "current speed" to be 100/full. Both sides are now optimally configured for 100/full operation.
- If either side reverts back to auto-negotiate for any reason, it will show the negotiated speed/duplex to be 100/half, which is a duplex mismatch and must be corrected.
- Following the instructions in section 2.1, heading "Speed/Duplex," examine the error counters on the Ethernet switch port and verify that the link is healthy (no errors).

S87xx/S8500 802.1p/Q and DSCP

On a Multi-Connect system, the port network control traffic between the S87xx/S8500 server(s) and IPSI boards traverses a closed control IP network. On this network there is no need to configure QoS, because all traffic is port network control traffic and has equal priority. QoS is required when there is the potential for contention for resources such as bandwidth, queue space, and processing power between various classes of traffic. This does not apply on the control IP network.

On an IP-Connect system the port network control traffic traverses the enterprise IP network, which services various classes of traffic. If QoS is desired and properly configured on this network, it may be necessary to have the S87xx/S8500 server(s) tag/mark the port network control traffic. This is only required on the interfaces that communicate with IPSI boards, as they are the only ones that participate in real-time IP telephony. Traffic is tagged/marked from these interfaces on a per destination basis for each IPSI board, as administered on the SAT **ipserver-interface** form (see section 3.4, heading "IP Server Interface Board"). For the 802.1p priority from the SAT form to be applied to the S87xx/S8500 server(s), L2 tagging must be enabled on the appropriate server interfaces via the **Configure Server – Configure Interfaces** web admin screen. The interfaces that communicate with IPSI boards have this option, and the others do not. The VLAN ID is always 0 for the S87xx/S8500 servers (follow the instructions in section 2.3, heading "Rules for 802.1p/Q Tagging").

3.2 S8300 Server

The S8300 is a Red Hat Linux server platform, similar to the S87xx/S8500, but on a compact media module that fits into a G700/G350/G250 gateway (always in media module slot 1). The S8300 is similar to the S87xx/S8500 in many ways. It is configured and managed via the same web interfaces, and, as with the other servers, there is little or no need for a customer to access the Linux shell.

In a G700 the S8300 must have an IP address on the same IP subnet as the MGP, with the same mask and default gateway (see G700 section below). This is because all media module slots in a G700 inherit the VLAN of the MGP, and therefore all VoIP media modules and the S8300 must be on the same IP subnet as the MGP. In a G350/G250 a VLAN must be designated as the ICC VLAN, and the S8300 must have an IP address on the IP subnet pertaining to that VLAN (see G350 section below).

An S8300 server can be in one of two modes: internal call controller (ICC) or local survivable processor (LSP). In ICC mode the S8300 is a standalone call server. In LSP mode it is a backup to the primary call

server and must be activated. An LSP does not accept station registrations or assume call processing responsibilities until it becomes active, which occurs when a gateway registers to it.
The S8300 SAT interface may be accessed using Avaya SA (ASA) or by telnet-ing to port 5023: **telnet <S8300 address> 5023** or SSH-ing to port 5022**.**. This could also be done by telnet-ing to the S8300 and typing **sat** from the Linux shell. S8300 ICC permits SAT configuration (changes and displays), but S8300 LSP does not (displays only) because it receives its Avaya Communication Manager translations from the primary server.
The S8300 connects to the G700/G350/G250 via a backplane 100M Ethernet interface, which is not configurable.

3.3 G700/G350/G250 Media Gateways

G700 P330/C360 L2 Switch

The P330 L2 switch is the base platform for the G700. All other logical/physical IP components (MGP, VoIP media modules, LSP) are connected to the P330 L2 switch. The asynchronous port (9600/8/N/1) marked **CONSOLE** on the face of the G700 connects the user to the P330 CLI. The IP expansion slot on the lower left corner of the chassis accepts the same X330 expansion modules used by the P330 switch. The most common ones are probably the WAN router module and the 16-port Ethernet module. The two Ethernet ports marked **EXT1** and **EXT2** are L2 switch ports. There is also an **Octaplane®** slot on the back of the chassis, just like the P330. For practical purposes the L2 switching portion of the G700 is equivalent to a 2-port P330 switch, which has a CLI similar to Cisco's CatOS and is configured using various **set** commands.
Three components of the P330 should be configured: the **inband** management interface, the **default route**, and the **switch** itself. The inband management interface is displayed and configured using the commands **show interface inband** and **set interface inband**, respectively. The inband interface requires a VLAN, an IP address, and a mask. The VLAN can be any of the VLANs active on the P330, and the IP address and mask must correspond to the IP subnet associated with that VLAN.
Once configured, the inband interface should be thought of as a host attached to the P330. This may seem non-intuitive, because the inband interface is the P330 and the way to administer the P330 remotely. However, like most L2 switch management interfaces, the inband interface is associated with a specific VLAN. As such, it is accessed just like any other host attached to the switch on a given VLAN – either directly from another host on the same VLAN/subnet, or by routing to it from a host on a different VLAN/subnet. Many mistakenly think that any host attached to the P330 should be able to access the inband interface directly, and this is not necessarily true. Hosts on different VLANs/subnets must route to the inband management interface via a L3 router.
Like any other IP host, the inband interface needs a default route if it is to route off of its VLAN/subnet. The default route for the inband interface is displayed and configured using the commands **show ip route** and **set ip route**, respectively. If there is more than one router on the inband VLAN/subnet, the inband interface may have additional routes based on destination subnets or hosts. These are displayed and configured using the same commands.
Finally, the P330 L2 switch itself has various configuration parameters, such as Spanning Tree, VLANs, trunking, and speed/duplex. These are configured just like on the P330 switch (see appendix E).

G700 Media Gateway Processor (MGP)

The media gateway processor (MGP) is the media gateway portion of the G700. The MGP manages the various media modules inserted into the G700. These media modules include analog port modules for analog phones, DCP port modules for DCP phones, DS1 modules for TDM trunks, and others. The media module associated specifically with IP telephony is the VoIP module. Each VoIP module is equivalent to a MedPro board and has 64 audio resources. A single VoIP module is built in to the MGP, and external VoIP modules can be added as necessary.

Like the P330 inband management interface, the MGP should be thought of as a host on the P330 L2 switch. The command **session mgp** from the P330 CLI puts the user into the MGP CLI. The MGP requires a VLAN, IP address, and mask. These are displayed and configured using the MGP CLI commands **show interface mgp** and **set interface mgp** (type **configure** to enter configuration mode for the **set** commands). The MGP may be on the same VLAN as the inband interface, or on a different VLAN. If on a different VLAN, a L3 router is required to route between the two VLANs. Like the inband interface, the MGP also needs at least a default route to route off of its VLAN/subnet. The MGP CLI commands are **show ip route mgp** and **set ip route** to display and configure MGP routes.
Each VoIP media module also requires an IP address using the **set interface voip v#** command. The VoIP modules inherit the VLAN, mask, and configured routes of the MGP, so there is no need to explicitly configure them for each VoIP module. The internal VoIP module is **voip v0**. An external VoIP module would be **voip v1** or **voip v2** or **voip v3** or **voip v4**, depending on which slot it is in. **show mm** shows all the media modules and their slot numbers.

G700 802.1p/Q and DSCP

The G700 can receive its audio and call signaling priority values from the call server's **ip-network-region** form or from local configuration. The MGP CLI command **show qos-rtcp** shows the locally set values and the values downloaded from the call server, along with which set of values is in effect. The command **set qos control** determines which set of values is used. The simplest implementation is to use the values from the call server. If configured locally, the **set qos** commands are used to administer the settings. There is no need, and no parameter, to set the VLAN ID because the MGP is already assigned to a VLAN via the **set interface mgp** command, and all modules inherit that VLAN ID.

G700 in Octaplane Stack vs. Standalone

The G700 can be placed within an Avaya P330 Octaplane stack, which provides 4Gig/full-duplex uplinks between the Avaya switches in the stack. There are pros and cons to this. The pros are that the entire stack can be managed as one unit via a single IP address, there is abundant bandwidth between the switches in the stack, and the entire stack can be uplinked to other network equipment without uplinking each individual switch in the stack. The cons are that the initial configuration can be a little more complex, and a problem associated with the stack can adversely affect the G700. Many factors can drive the decision to use or not use the Octaplane.
Device and uplink management are key factors. If several G700 gateways are co-located in the same rack, it makes practical sense to use the Octaplane stacking feature. This allows the P330 components of all the G700s to be managed via a single inband interface. But more importantly, it eliminates the need for each G700 to be uplinked to the next network device individually.
When determining whether or not a single G700 should be added to an existing Octaplane stack of P330 switches, the relative importance of the G700 to the other devices is another factor. A G700's primary role is that of IP telephony, specifically media conversion. A P330 switch's primary role is that of L2 switching – processing and forwarding Ethernet frames, managing broadcast domains (VLANs), participating in Spanning Tree, etc. Depending on the implementation, especially if there are no dependencies between the G700 and the P330 stack, it may be prudent to keep the two roles separate so that a problem with either the G700 or the P330 stack does not adversely affect the other. These points are mentioned to provoke thought in design and implementation. Whatever the decision, a G700 can fully participate in Octaplane stacks, with other G700s or with P330 switches.
Bandwidth is another key factor for using or not using the Octaplane stack. The G700 components (P330 inband, MGP, VoIP modules, S8300) require a certain amount of bandwidth to communicate off the chassis. Each VoIP module consumes a maximum of approximately 6Mbps to service 64 G.711 calls using 20-ms packets. With up to five VoIP modules on a single G700, the maximum bandwidth consumption is approximately 30Mbps. Other than firmware and translation downloads the bandwidth

consumed by the other components is negligible. Therefore, a single 100M uplink from EXT1 or EXT2 to another Ethernet switch is sufficient for the G700 itself.
The added bandwidth of the Octaplane stack might be required when the 16-port X330 Ethernet expansion module is used in the G700, and the hosts attached to that module communicate mostly to other hosts not on the G700. If the hosts on the expansion module are IP telephones, a 100M uplink is sufficient. But if PCs are attached to the phones and the PCs frequently communicate off the G700, a 100M uplink may not be sufficient.

G350 Media Gateway

The G350 is similar to the G700 in many ways. Therefore, this section details the differences while referring to the G700 explanations for similarities.
Two significant differences between the G350 and G700 are capacity and architecture. The G350 supports much fewer users (~40 max) than the G700 (~450max). As such, the G350's internal VoIP module has only 32 audio resources, as opposed to 64 in the G700's internal VoIP module, and in the external VoIP media module and MedPro board. The G350 also cannot presently accept external VoIP modules.
The primary architectural difference between the G350 and G700 is that the G350 is an integrated platform. The L2 switch, MGP, and internal VoIP module all share the same processing engine and same IP address. In addition, a L3 router is integrated into the G350, whereas the G700 can accept a L3 router as an expansion module. The resulting G350 CLI has two components. The L2 switch and MGP commands are practically the same as on the G700, using **set** commands similar to the P330 switch and Cisco's CatOS. The L3 router commands are practically the same as on the X330WAN and P330R routers, using commands similar to Cisco's IOS. The G350 utilizes both command sets in a single CLI.
The G350 can have several physical and logical interfaces, and there is no single inband interface as on the G700's P330 switch component. One of the G350's IP interfaces must be designated the primary management interface (PMI). There is a default designation, but it can be changed by inserting the command **pmi** under the desired interface. The PMI, among other things, is the interface used by the MGP and internal VoIP module. This means that H.248 media gateway signaling and RTP audio are sourced by and terminated on the PMI. When an S8300 media server is inserted into a G350, one of the VLANs on the G350 must be designated the ICC VLAN. This is done by inserting the command **icc-vlan** under the desired VLAN interface.
The remaining G350/G700 similarities and differences are as follows. Refer to the G700 sections above for more details on each subject.

- Like the G700, the G350 needs at least a default route to route off of its connected VLANs/subnets. But there is a single default route for the whole unit, as opposed to a route for the P330 component and a route for the MGP component. This is configured using the **ip route** command.
- The G350 and G700 L2 switch configurations (Spanning Tree, VLANs, trunking, speed/duplex) are very similar.
- The G350 and G700 802.1p/Q and DSCP configurations are very similar.
- The G350 does not have an Octaplane interface.

G250 Media Gateway

The G250 is very similar to the G350, but with less capacity – ~10 users max. The fundamental configuration requirements for the G250 are essentially the same as for the G350, as described above. Like the G700 and G350, the G250 is capable of housing an S8300 server in ICC or LSP mode. In addition, unlike the other two gateways, the G250 can act as a survivable H.323 gatekeeper. This feature is called Standard Local Survivability (SLS), and it allows the G250 to be a call server with very limited features when communication to the primary call server is lost. For simplicity, SLS can be thought of as an integrated LSP with very limited features. The details of SLS and its configuration are not covered in this document.

General Guidelines Related to Gateways

The MGC List, Transition Point, and Primary Search Time must be configured properly on all gateways. These are the parameters that determine which devices are primary controllers, which is the LSP, and when to fail over to the LSP. There may be undesirable behaviors if these parameters are not configured properly, such as a gateway registering with an LSP when a primary controller is available; a gateway registering with an LSP too soon after an outage; and different gateways at the same location registering with the LSP at different times. See the "Avaya Communication Manager Network Region Configuration Guide" at www.avaya.com for detailed instructions on configuring these parameters.

When connecting a gateway to another Ethernet switch, the uplink between the two switches should be fixed at 100M/full-duplex (see section 2.1, heading "Speed/Duplex"). Furthermore, if 802.1p/Q tagging information – VLAND ID and/or L2 priority – is to be passed across the uplink, both switches must have 802.1Q trunking enabled with matching VLANs on the connected ports.

For survivability reasons at remote offices, the LSP, media gateway, and local IP phones should all be on the same voice VLAN/subnet. If the LSP, media gateway, and IP phones are on different subnets, they depend on a router or routing process to function. This is not desirable, especially at a branch office where there is typically only one router. The IP telephony system should be able to function even if that router or routing process fails. In larger remote offices it may not be feasible to put all VoIP endpoints on a single subnet, but the principle of minimizing dependencies still holds.

For remote offices where the WAN link terminates on the gateway, whether on the X330WAN router or natively on the G350/G250 itself, the DSCP values for audio and signaling must be 46 and 34 respectively. The X330WAN router in "voip-queue" mode and the G350/G250 gateways are optimized to use these values for QoS on the WAN link. These values can be configured locally via the **set qos bearer/signal** commands, or they can be downloaded from Communication Manager. On Communication Manager these values are configured on the SAT **ip-network-region** form for the region to which the gateway is assigned. A gateway is configured to use the Communication Manager values by executing **set qos control remote** on the MGP/gateway CLI. The CLI command **show qos-rtcp** displays the locally set and remotely downloaded values, as well as which values are in use.

The G700, G350, and G250 gateways are all administered in Communication Manager via the SAT **media-gateway** form, which is covered in section 3.5.

3.4 G650/G600, MCC1, and SCC1 Gateways (Port Networks)

The G650/G600, MCC1, and SCC1 are non-H.248 media gateways. They are controlled via the Avaya CCMS protocol, unlike the G700/G350/G250 gateways which are controlled via the H.248 protocol. The CCMS-based gateways are better known as port networks, and they share the same port boards. The most significant boards related to IP telephony are the C-LAN (TN799DP), MedPro (TN2302AP), and MR320 (TN2602AP) boards. Boards with these specific codes are required for Communication Manager; previous board revisions cannot be used.

C-LAN Capacity and Recommendations

The Control LAN (C-LAN) board is the IP interface for many functions, including H.225 call signaling for IP stations and IP trunks, H.248 media gateway control signaling, connectivity to various adjuncts, and SAT administrative access via TCP/IP. Each connection for one of these functions requires at least one TCP socket on the C-LAN board. The C-LAN board can support over 400 sockets under light usage conditions. However, with heavier usage comes greater load on the CLAN and worse performance. Furthermore, regardless of usage it is highly discouraged to operate the C-LAN near maximum capacity in a production environment. The following conservative recommendations are offered for typical environments, and can vary based on usage levels.

- Adjuncts such as CMS, CDR, AUDIX® Messaging System, and others should be placed on separate C-LAN boards that are not used for call signaling or media gateway control signaling. This is

common practice primarily due to business impact of the adjunct and the consequent need to isolate the adjunct, as well as to quickly troubleshoot any problems related to the adjunct.
- In a typical call center environment, design for a normal operating load of 200-250 IP stations plus 6 media gateways per C-LAN.
- In a typical non-call center business environment, design for a normal operating load of 250-300 IP stations plus 8 media gateways per C-LAN.
- The number of signaling groups (IP trunks) per CLAN depends greatly on the configuration and usage of each signaling group. Configuring the signaling group to have "calls share IP signaling connection" – an option between Avaya systems – requires less C-LAN resources than if each individual call has its own connection. The greater the usage of the signaling group (frequency of calls, features utilized during calls, number of simultaneous calls, etc.), the greater the C-LAN resource consumption. As a very general rule based on anecdotal evidence of typical IP trunk usage, and assuming calls share IP signaling connection, substitute one signaling group for ten IP stations in the two preceding bullet items. Another option is to dedicate C-LAN boards and network regions specifically for IP trunks.

C-LAN and MedPro/MR320 Protocols and Ports

Call signaling and media conversion between analog, TDM, and IP are key IP telephony functions. The S8700, S8500, and DEFINITY servers use distributed C-LAN boards to front-end the call signaling, and distributed MedPro/MR320 boards to perform the media conversion. The following table lists the protocols and ports used by both boards. Section 3.5, heading "ip-network-region" gives instructions on how to configure the MedPro/MR320 UDP port range. *See Appendix D for guidelines on configuring access lists.*

C-LAN	UDP 1719	H.225 RAS – IP station registration
	TCP 1720	H.225 Q.931 – call signaling for IP stations and IP trunks
	TCP 2945	H.248 media gateway control signaling
	TCP 1039	H.248 encrypted MG control signaling
MedPro/ MR320	UDP 2048 – 65535 (configurable)	RTP-encapsulated audio

Table 6: C-LAN and MedPro/MR320 protocols and ports

C-LAN and MedPro/MR320 Network Placement

Place the C-LAN and MedPro/MR320 boards on highly reliable subnets as close as possible to the majority of IP endpoints (ie, IP phones and softphones). Keep in mind that both call signaling and audio from all IP endpoints require these boards. Therefore, it may not be good practice to place these boards on a subnet containing many enterprise resources – such as a server farm – where there is heavy traffic both on the subnet and on the uplink(s) to the subnet. On the other hand, a server farm is typically where the most reliable and redundant network resources are deployed. A thorough understanding of the network and network traffic is required to ultimately determine the best placement of these critical boards.

C-LAN and MedPro/MR320 Speed/Duplex

Use the SAT **ip-interface** form to configure the speed and duplex for the C-LAN and MedPro/MR320 boards. It should be standard procedure to properly set the speed and duplex on all C-LAN and MedPro/MR320 boards, and to configure the associated Ethernet switch ports accordingly. This results in much better system stability and audio quality than if the boards and Ethernet switch ports are left to auto-negotiate. See section 2.1 under the "Speed/Duplex" heading.

The default speed/duplex setting on the MedPro/MR320 board is auto-negotiate. The default speed/duplex setting on the TN799DP C-LAN board is 10/half, to make it backwards compatible with the previous TN799C board, which could only do 10/half. When a C-LAN or MedPro/MR320 is inserted into one of the port networks, the board receives its speed/duplex programming from Communication Manager, per the appropriate form. If for any reason a board loses this programming, it reverts back to the board's default.

The maximum MedPro throughput is 5.8Mbps for 64 G.711 20-ms calls.

The maximum MR320 throughput is 29Mbps for 320 G.711 20-ms calls.

The maximum throughput for a MedPro board is 5.8Mbps, which is what is required for 64 G.711 20-ms calls over Ethernet. The maximum throughput for a C-LAN board is much less than this. Therefore, the minimum speed/duplex requirements are 100/half for the MedPro and 10/half for the C-LAN. Due to its high capacity, the MR320 board should always run at 100/full.

If there is poor audio quality on calls going through a particular MedPro/MR320 board, follow these steps to determine if a speed/duplex mismatch between the MedPro/MR320 and the Ethernet switch is the cause.

- Check both the board (**get ethernet-options <slot #>**) and the Ethernet switch port and verify that they are set to the same speed/duplex or have auto-negotiated to the same speed/duplex.
- Check for L1 errors as instructed in section 2.1 under the "Speed/Duplex" heading.
- Send a continuous ping (**ping -t**) to the MedPro/MR320 from a Windows machine. If the pings intermittently fail and the failures coincide with periods of poor audio quality, then there is probably a speed/duplex problem between the board and the Ethernet switch.

C-LAN and MedPro/MR320 802.1p/Q and DSCP

See section 3.5, headings "ip-interface" and "ip-network-region." L2 and L3 prioritization on the C-LAN requires the TN799DP board with firmware v5 or later.

MR320 Capabilities and MR320 Bearer Duplication

The TN2602AP IP Media Resource 320 provides either 80 or 320 encrypted or unencrypted channels of 2-way audio RTP streams or conversations. Channel capacity is dependent on software licensing. The MR320 supports G.711, G.729A/B and G.726A codecs. See Table 6 for a comparison of Medpro and MR320 capabilities. A single port network can have up to two TN2602AP circuit packs only. As result, the port network can have either two duplicated TN2602AP circuit packs or two load balancing TN2602AP circuit packs, but not both a duplicated pair and a load-balancing pair. However, in a Communication Manager configuration, some port networks can have a duplicated pair of TN2602AP circuit packs and other port networks can have a load-balancing pair of TN2602AP circuit packs and some port networks can also have single or no TN2602AP circuit packs.

The TN2602AP IPMedia Resource 320 can provide duplicated bearer for IP Connected Port Networks. This enables customers to administer IP-PNC with critical bearer reliability. A port network supports a maximum of two TN2602AP circuit packs and they can be administered for duplication. Duplicated TN2602AP circuit packs will operate in an Active-Standby mode. State of health parameters exist between the two boards to determine when it is appropriate to interchange duplicated TN2602AP circuit packs. The failover from Active to Standby can take up to 3 seconds, depending on the type of fault, without interruption of service. Duplicated TN2602AP circuit packs in a PN share a virtual IP and virtual MAC address. These virtual addresses are owned by the currently-active TN2602. In addition to the virtual IP address, each TN2602 has a "real" IP address. All bearer packets sent to a PN that contains duplicated TN2602AP circuit packs, regardless of whether the packets originate from TN2602s in other PNs or from IP phones or gateways, are sent to the virtual IP address of the TN2602 pair in that PN. Whichever TN2602AP circuit pack is active is the recipient of those packets. When failover to the

standby TN2602 occurs, a negotiation between TN2602s to determine which TN2602 is active and which is standby takes place. State-of-health, call state, and encryption information is shared between TN2602s during this negotiation. The newly-activeTN2602AP circuit pack sends a gratuitous address resolution protocol (ARP) request to ensure that the LAN infrastructure is updated appropriately with the location of the active TN2602 Other devices within the LAN will update their old mapping in ARP cache with this new mapping. It is also possible to invoke an interchange manually via a software command.

Duplicated TN2602AP circuit packs must be in the same subnet. In addition, the Ethernet switch or switches that the circuit packs connect to must also be in the same subnet. This shared subnet allows the Ethernet switches to use signals from the TN2602AP firmware to identify the MAC address of the active circuit pack. This identification process provides a consistent virtual interface for calls.

The Communication Manager license file must have entries for each circuit pack, with the entries having identical voice channels enabled. In addition, both circuit packs must have the latest firmware that supports bearer duplication.

Capability	TN2302AP (HW Version 11 or later)	TN2602AP
Codecs	• G.711 64 maximum unencrypted channels, 48 maximum encrypted channels. • G.729B and G.723.1 32 maximum unencrypted, 24 maximum encrypted channels.	• G.711 - 80 or 320 channels by license, unencrypted or encrypted • G.729A, G.729AB - 80 or 320 channels by license, unencrypted or encrypted • G.726A - 80 or 320 channels by license, unencrypted or encrypted.
Fax Relay (proprietary)	16 unencrpyted, 12 encrypted	80 or 320, by license, unencrypted or encrypted
Fax T.38/Modem relay	16 unencrpyted, 12 encrypted	N/A
Fax/Modem Pass-thru	64 G.711, 32 G.729	80 or 320 by license
TTY Relay	32 G.729 unencrypted, 24 encrypted	80 or 320 by license
TTY Pass-thru	32 G.729 unencrypted, 24 encrypted	80 or 320 by license
Echo Tail	32 ms tail	128 ms tail, 24 ms moving window
SSH/SCP Support	No	Yes
Active-Standby Failover	No	Yes

Extreme Measures for MedPro and Other IP Boards on Cisco Switches

This information is intentionally placed here and not in section 2.1, because it is a last-resort measure. On rare occasions a MedPro board's Cisco switch port may flap up and down continuously. This is manifested by bridge join/leave messages for CatOS-based switches, and interface up/down messages for IOS-based switches. Sometimes this problem is caused by the backplane I/O cable not being Cat5 compliant, and Avaya Tier 3 support can determine whether or not this is the case. Sometimes this problem is a compatibility issue between the MedPro and the Cisco switch. After the instructions in section 2.1, headings "Ethernet Switches" and "Speed/Duplex" have been followed, if the Cisco switch port continues to flap up and down, consider the options described in the next paragraph.
The Cisco white paper "Troubleshooting Cisco Catalyst Switches to Network Interface Card (NIC) Compatibility Issues [4 p.6]" describes the flapping problem mentioned above and offers a suggestion to

adjust the jitter tolerance (not related to audio jitter) on Cisco switches. The CatOS global command (which is hidden) is **set option debounce enable** (**disable** to undo). This command increases the jitter tolerance to 3.1 nsec from the 1.4-nsec default. The IOS interface command is **carrier-delay 4** (**no carrier-delay** to undo). This adjusts the carrier transition delay to 4 seconds. If these commands do not correct or improve the flapping condition, put the switch back to its original state and try operating at 10/half until the problem can be resolved.

IP Server Interface (IPSI) Board

The IP Server Interface (IPSI) board is installed in a G650/G600, MCC1, or SCC1 port network, and it is the port network's interface to communicate with the call server(s). Most of the programming for an IPSI board is done on the SAT **ipserver-interface** form, which has commands **change ipserver-interface #**, **display ipserver-interface #**, and **list ipserver-interface**.
If IP Control is 'y' the board is acting as an IPSI; otherwise ('n') it is acting as a tone clock. The 'n' option is primarily used for migrating a non-IPSI port network to an IPSI port network. Ignore Connectivity in Server Arbitration has to do with whether or not connectivity to this IPSI is factored into the decision to interchange S87xx servers. In most cases this is set to 'n', but in rare cases it could be set to 'y' for IPSIs in remote locations with poor network connectivity back to the servers. The intent would be to avoid server interchanges caused by frequent and inconsistent loss of communication to this IPSI. Location is the board slot #. Host is the board's static IP address if configured manually, or the hostname if the address was obtained via DHCP. DHCP ID is the hostname. Socket Encryption, if the parameter is present, allows the control link between the IPSI and call server to be encrypted. When QoS is enabled the 802.1p and DiffServ parameters contain the values to be applied to the call server when communicating with this IPSI board (values are not applied to the IPSI board itself).
The IPSI's speed/duplex and L2/L3 priority values are configured on the board itself, instead of via SAT forms. From the IPSI board type **ipsilogin** at the **[IPSI]:** prompt, and enter the login name and password to access the **[IPADMIN]:** prompt. The commands to display and configure the control port speed and duplex are **show port 1**, **set port negotiation 1**, **set port speed 1**, and **set port duplex 1**. The commands to display and configure the L2 and L3 priority values are **show qos**, **set vlan tag**, **set vlan priority**, and **set diffserv**. Be sure to understand what these values do before setting them (see all of section 2.3, particularly the heading "Rules for 802.1p/Q Tagging").

3.5 General IP-Telephony-Related Configurations (SAT Forms)

The SAT interface has various "forms" that are used to configure specific features. This section covers the forms used to configure general IP telephony. Most of the forms have a **display** option to view the current configurations, and a **change** option to change them. Some also have a **list** option to view, for example, a broad list of stations without seeing in detail how each station is configured.

ethernet-options

As of Avaya Communication Manager 2.0 each IP board's speed and duplex settings are configured using the **ip-interface** form. The **ethernet-options** form has the **list** and **get** options to verify actual speed/duplex settings against configured settings for all boards and individual boards respectively. With each new system or IP board installation, one standard procedure should be to apply matching speed/duplex settings to each IP board and its corresponding Ethernet switch port.

node-names ip

Options are **change** and **display**. This form is used to define arbitrary names and associate an IP address with each name. For example, the name "c-lan_80" could be defined to describe a C-LAN board on the 80 subnet with address 192.168.80.10, and the name "medpro_80" could be defined to describe a MedPro board on the 80 subnet with address 192.168.80.11.

ip-interface

Options are **change**, **display**, and **list**. This form is used to configure individual IP boards. The first step is to associate a board Type and Slot # to a previously defined Node Name, and to give the board a Subnet Mask and default Gateway and assign it to a Network Region. For example, the board type C-LAN in slot 01A05 can be associated with the node name "c-lan_80" defined earlier. This assigns the IP address 192.168.80.10 to the C-LAN board in slot 01A05. Then the board can be given the mask 255.255.255.0 with default gateway 192.168.80.254. The board can also be assigned to network region 1.

802.1p/Q tagging for an IP board is also enabled or disabled on this form. A number (including 0) in the VLAN field indicates the VID, and it means that tagging is enabled on the board with that VID. Although most implementations where tagging is enabled should use VID 0, other VIDs are permitted as well. The letter 'n' in this column means that tagging is disabled on the board, and a blank means that tagging is not supported on the board. To properly enable L2 tagging on the C-LAN and MedPro/MR320 boards, follow the instructions in section 2.3 under the heading "Rules for 802.1p/Q Tagging."

The speed and duplex settings for an IP board are configured on this form under the Ethernet Options heading.

The TN2602AP MR320 board has a VOIP Channels parameter to indicate how many channels are active on the board. While this parameter is configurable, it is restricted by licensing. The initial licensing options are to purchase a number of boards with 80 channels each, and a number of boards with 320 channels each.

The 2602AP MR320 board can be administered for Critical Reliable Bearer.

The Shared Virtual Address is the virtual ip-address owned by the currently active MR320 and must be administered in the same subnet as the "real" ip-addresses. The duplicated MR320s also share a virtual MAC address that is automatically assigned by one of four virtual MAC tables. Each Virtual MAC table contains 64 cached AVAYA owned MAC addresses and each table can be displayed with display virtual-mac-table SAT command.

The C-LAN board parameter Number of CLAN Sockets Before Warning. This is related to the information in section 3.4, heading "C-LAN Capacity and Recommendations." This parameter only dictates when a warning is triggered and does not affect the total number of TCP sockets supported by the C-LAN. Although the recommended number of sockets on a C-LAN may be less than 400, it is advisable in many cases to wait until 400 (default value) to trigger an alarm.

The parameter Receive Buffer TCP Window Size should be left at the default value of 8320. The default value should only be changed by AVAYA Services. The Allow H.323 Endpoints and the Allow H.248 Endpoints fields are administered to allow or disallow registration of endpoints and gateways on the C-LAN. The Gatekeeper Priority parameter is used for Alternate Gatekeeper lists and is available when H.323 endpoints are allowed to register. The lower the number the greater the priority.

data-module

Options are **change**, **display**, and **list**. This form is used to assign an extension (required for call processing) to a C-LAN board, and to specify other parameters. The Extension can be any valid extension in the dial plan, and does not have to be a DID extension. The Type is Ethernet. The Port is the board slot # appended with the number 17 (ie, 01A0517). The Link number can be any available number from the output of the **display communication-interface links** command. The Name is the previously defined node name (ie, "c-lan_80").

ip-codec-set

Options are **change**, **display**, and **list**. This form is used to define the codec sets that are referenced by other IP telephony forms. Up to 7 codec sets may be defined with 5 codecs, listed in order of preference, in each set. G.711 (uncompressed) and G.729 (compressed) are the recommended codecs for LAN and WAN, respectively. No silence suppression and 20-ms voice packets are also recommended.

A word of caution: CM allows for the administration of the G.726A codec type but it is only available on the MR320 (TN2602). The TN2302 does not support G.726A.

Note about silence suppression: Although silence suppression conserves bandwidth by not transmitting audio packets during periods of silence, its use typically results in audio clipping, which most users consider unacceptable. The G.729B codec may be a better alternative to silence suppression. Rather than not transmitting during silence, this codec transmits silence in a condensed format that requires less bandwidth. The audio quality of G.729B is still noticeably inferior to G.729.

Larger packet size = less bandwidth
Smaller packet size = more bandwidth

Larger packet size → low loss, high jitter network
Smaller packet size → high loss, low jitter network

20-ms packet size recommended

Note about voice packet size: Audio is encoded in increments called frames, with the typical frame size being 10ms. The packet size, or number of frames per packet, is a measure of how much audio is sent in each IP packet. Experience has shown that a 20-ms packet is a good compromise between audio quality and bandwidth consumption. Reducing to 10ms doubles the number of packets put onto the network, but only 10ms of audio can be lost when a packet fails to reach its destination or arrives out of order. Going beyond 20ms reduces the number of packets put onto the network, but there is greater potential for poor audio quality when there is high packet loss.
Larger packets work better in low loss, high jitter networks. Smaller packets work better in high loss, low jitter networks. 20-ms packets are a good compromise.

The Media Encryption portion of this form is an ordered list of preferred media encryption options. For example, an ordered list of AES, AEA, and none means that AES encryption is preferred first, then AEA encryption if AES is not possible, then no encryption if neither AES nor AEA is possible. This list may contain one or more items.

Allow Direct-IP Multimedia has to do with video over IP, which is beyond the scope of this document. For information on the remaining FAX, Modem, TDD/TTY, and Clear-channel parameters, see the product documentation "Administration for Network Connectivity for Avaya Communication Manager" (555-233-504), chapter 3, heading "Administering FAX, modem, TTY, and H.323 clear channel calls over IP trunks." See also the document "Avaya FoIP, MoIP, & TTYoIP" at www.avaya.com.

ip-network-region

Options are **change**, **display**, and **list**. This form is used to define the characteristics of an Avaya Communication Manager network region. While this section describes the configuration parameters of the **ip-network-region** form, the overall explanation of network regions and guidelines for network region design are covered in detail in the "Avaya Communication Manager Network Region Configuration Guide" at www.avaya.com.

The Location parameter is used to assign IP stations in this network region to a specific geographic location identifier.

The Authoritative Domain applies to Session Initiation Protocol (SIP) applications, which are not covered in this document.

The Name is an arbitrary string to describe the network region.

The Codec Set refers to one of the seven codec sets defined using the **ip-codec-set** form, and specifies which codec(s) are used by the endpoints in this network region.

The UDP Port Min/Max is the range used for RTP audio by the MedPro and MR320 boards and VoIP media modules in this network region. Use the following points to configure a more narrow UDP port range (to set up security filters, for example).

- 2048 is the beginning of the range by default, but this can be changed to a higher starting point. It is recommended to use UDP ports outside the range of reserved ports. A starting port of 50000 is outside the range of any reserved ports.
- The MedPro supports 64 uncompressed audio streams (G.711 codec) or 32 compressed audio streams (G.729 codec) or any combination using the following formula: [uncompressed streams + 2(compressed streams)] = 64. The MR320 supports up to 320 audio streams, depending on licensing and configuration.
- Per the RTP standard, each audio stream requires an even-numbered UDP port for the RTP audio, and the subsequent odd-numbered UDP port for the RTCP control exchange.
- Therefore, to support X audio streams the UDP port range must contain 2X consecutive ports, beginning with an even port and ending with an odd port. Since the absolute maximum value for X is 320 (MR320 board), the largest required UDP port range is 640. Duplicated Media Resource 320 (MR320) boards need 320x4 UDP ports or 1280 ports. In this case a port range of 50000 to 51279 can be administered.

The DiffServ (DSCP) and 802.1p/Q parameters are the L3 and L2 priority values for call signaling from C-LANs in this network region, and audio from MedPros/MR320s in this network region. The L2 values are only applied to boards that have L2 tagging enabled via the **ip-interface** form. The reason for the two forms is that L2 tagging and VID can vary per board across a network region, but the priority values are typically uniform throughout the region.
Ideally two different sets of L2/L3 values should be specified for signaling and audio. However, for practical purposes in many applications it is common to use the same set of values for both signaling and audio. Appendix F gives examples of how the L3 values are used in conjunction with QoS on routers. L2 and L3 prioritization on the C-LAN requires the TN799DP board with firmware v5 or later.

Direct IP-IP Audio (shuffling) and IP Audio Hairpinning within a network region and across different network regions are enabled and disabled on this form. Direct IP-IP audio permits calls between IP endpoints to "shuffle" directly to each other, instead of speaking through the MedPro/MR320 board or VoIP module. If a feature that requires the media gateway, such as conferencing, is activated during the call, the endpoints shuffle back to the MedPro/MR320 board or VoIP module. If the conference ends and only two parties remain, the IP stations shuffle back to one another.

Hairpinning permits calls between IP endpoints to speak through the MedPro/MR320 board or VoIP module, but without any transcoding. This is essentially a relay feature for IP endpoints that are not capable of redirecting their audio streams. None of the Avaya IP telephones have this limitation.

Direct IP-IP Audio and IP Audio Hairpinning are generally enabled, unless there is an Avaya R300 or MultiVOIP gateway in this network region, in which case hairpinning should be disabled. Also, for direct IP-IP audio to function across different network regions, an inter-region codec set must be specified and the regions must be connected via the inter-region connectivity matrix beginning on page 3 of this form.

There are network address translation (NAT) options for direct IP-IP audio. Since Avaya Communication Manager 1.3, Avaya has permitted shuffling between endpoints that are separated by NAT. NAT has been a hurdle for VoIP due to the fact that the address in the IP header is translated, but embedded IP addresses in the H.323 messages are not translated. This hurdle has been overcome to some extent with the "NAT shuffling" feature in Communication Manager, without the need for H.323-aware NAT devices. See "NAT Tutorial and Avaya Communication Manager 1.3 NAT Shuffling Feature" at www.avaya.com.

Note: In addition to the **ip-network-region** form, shuffling and hairpinning must be enabled on two other forms: the **system-parameters features** form, page 16; and the **station** form, page 2, for each station.

The RTCP monitoring feature is used with the Avaya VoIP Monitoring Manager (VMM). Enabling this feature causes the audio endpoints in this region to send periodic RTCP reports to VMM. VMM uses these reports to keep a history of audio quality for all reporting endpoints. The default server parameters are configured on the **system-parameters ip-options** form. If the default settings are not desired in any given network region, specific settings can be applied on a per region basis.

The RSVP feature requires careful integration with the IP network and must not be enabled without the supporting IP network configurations. These configurations can be cumbersome and require a significant amount of network overhead. A better call admission control (CAC) mechanism is native to Communication Manager as of 2.0 and is explained in detail in the "Avaya Communication Manager Network Region Configuration Guide" at www.avaya.com.

The H.323 Link Bounce Recovery parameters, the LSP list on page 2 of this form, and the inter-region connectivity matrix beginning on page 3 of this form are covered in detail in a separate document. See the "Avaya Communication Manager Network Region Configuration Guide" at www.avaya.com.

Inter-Gateway Alternate Routing (IGAR) on page 2 of this form is a new feature for Communication Manager 3.0. This feature is covered in detail in the "Avaya Communication Manager Network Region Configuration Guide" at www.avaya.com. Related to IGAR is a new parameter on the **cabinet** form to assign the cabinet to a network region. The assignment of a cabinet to a network region, which is a concept new to Communication Manager 3.0, applies primarily to IGAR. It has no relation to IP boards in that cabinet, and it does not assign traditional resources attached to that cabinet, such as non-IP stations and trunks, to a network region.

ip-network-map

Options are **change** and **display**. This form is used to assign stations to Communication Manager network regions by IP address range or subnet. If a station's IP address does not fall into any of the ranges configured on this form, the station is assigned to the same network region as the gatekeeper it registers with. Whether by assignment on this form or by inheritance, it is very important to assign IP stations to the proper network region. To understand how these methods of network region assignment affect the station, see the "Avaya Communication Manager Network Region Configuration Guide" at www.avaya.com.

The VLAN column is used to send a VID to IP phones. This field should only be used if DHCP option 176 is not available. If such is the case, then two rows are required on this form: one row for the data VLAN through which the phone passes, and another row for the voice VLAN on which the phone finally resides, with both rows containing the voice VID. The resulting functionality would be as follows.

- IP phone boots and obtains address on data VLAN.

- IP phone registers with Communication Manager from data VLAN.
- **ip-network-map** shows phone assigned to a specific network region on a specific voice VLAN.
- Communication Manager directs phone to that voice VLAN.
- IP phone releases data VLAN address and obtains address on voice VLAN.
- IP phone registers with Communication Manager from voice VLAN.
- **ip-network-map** shows phone assigned to a specific network region on a specific voice VLAN.
- Communication Manager directs phone to that voice VLAN, but phone is already on it.
- Using this method, the phone applies the L2 priority values for audio and signaling as administered on the **ip-network-region** form for the phone's region. Using the recommended DHCP option 176 method, the phone applies the L2 priority values received from DHCP.

The Emergency Location Extension is part of the E911 features of Communication Manager and is not within the scope of this document.

station

Options are **add**, **change**, **display**, and **list**. This form is used to define stations. To specify an IP station the Type must be an IP model. The Port is automatically set to X for an IP phone when the station is first added. This is changed to S##### – an automatically assigned internal port number – when the phone registers with the call server. The IP Softphone inquiry is regarding whether or not a softphone is permitted to take over the extension. This field applies to non-IP stations as well, as an IP softphone can take over an analog or DCP extension and emulate that set type. Survivable GK Node Name provides an option for the station to fail over to an Avaya G150 Media Gateway or a MultiTech MultiVOIP gateway when no other gatekeeper is available. Direct IP-IP Audio and IP Audio Hairpinning for the individual station is configured on page 2 of this form.

trunk-group and signaling-group

Options are **add**, **change**, **display**, and **list**. These forms are used to define trunks, including H.323 IP trunks. This document is concerned only with the IP-specific configuration parameters.
On the **trunk-group** form, the Group Type should be isdn, the Carrier Medium should be IP, and each member's Port designation (beginning on page 3 of the form) should also be IP. Once the members are used for active calls the call server automatically changes the port designations to T#####, which are internal port numbers. The number of members determines the number of simultaneous calls.
The **signaling-group** parameters are as follows.

- Group Type: h.323.
- Remote Office: 'n' in most cases, 'y' if the far end is a G150, R300, or MultiVOIP gateway.
- Trunk Group for Channel Selection: Specify the trunk group configured as described above.
- Near-end Node Name: The node name of the local gatekeeper (C-LAN or S8300) terminating the H.323 signaling link, as defined in the local call server's **node-names ip** and **ip-interface** forms.
- Near-end Listen Port: 1720 by default. This is the default TCP port used by the gatekeeper for H.225 call signaling.
- Far-end Node Name: The node name of the far-end gatekeeper terminating the H.323 signaling link, as defined in the local call server's **node-names ip** form.
- Far-end Listen Port: 1720 by default if far-end gatekeeper is an Avaya server or Cisco Call Manager. May vary from device to device if configured to listen on a different TCP port.
- Far-end Network Region: The numeric identifier of the locally defined network region with which the far-end gatekeeper is associated. That is, the far-end gatekeeper is treated as if it were an endpoint in the locally defined network region specified in this field.
- RRQ Required: 'y' if the signaling group is for a G150, R300, or MultiVOIP gateway. This requires the gateway to send a RAS Registration Request to bring the signaling group into service.

- Media Encryption: New to Avaya Communication Manager 2.1. This parameter permits media encryption between the two Avaya systems joined by this IP trunk. Selecting 'y' invokes a passphrase, and both ends of the IP trunk must have the identical passphrase. This facilitates a key exchange between the systems, which makes media encryption possible between endpoints on the two systems, as long as the **ip-codec-set** forms on both systems are configured with matching encryption options. In other words, enabling encryption on the **ip-codec-set** form permits encryption within a system. Media encryption between two systems is possible when they have compatible codec sets and encryption options, and are connected by an IP trunk with this feature enabled.
- DTMF over IP: See the section below for the **system-parameters ip-options** form.
- Calls Share IP Signaling Connection: 'y' if the far-end is an Avaya device, 'n' if it is another vendor's device. 'y' means that a single H.225 signaling connection is used for all trunk members (all calls), and 'n' means that each trunk member (each call) uses a separate signaling connection. The G150, R300, and MultiVOIP gateway require this to be set to 'y'.
- Bypass if IP Threshold Exceeded: Part of a feature commonly referred to as "TDM fallback" or "IP trunk bypass." This parameter has to do with whether or not a TDM fallback trunk is utilized when the IP network fails or performs poorly between the near-end and far-end gatekeepers. The thresholds for this fail-over are configured in the **system-parameters ip-options** form, as described in Appendix G. Appendix G is a Q&A discussion on the IP trunk bypass feature and associated issues related to IP trunks.
- Direct IP-IP Audio Connections: 'y' typically, same as with endpoints.
- IP Audio Hairpinning: 'y', unless G150s, R300s, or MultiVOIP gateways can talk across the trunk.

The LRQ Required parameter allows IP trunk availability to be determined on a per call basis. When this option is enabled a RAS-Location Request (LRQ) message is sent to the far-end gatekeeper prior to each call over the IP trunk. The far-end gatekeeper responds with a RAS-Location Confirm (LCF) message and the call proceeds. The absence of an LCF from the far-end gatekeeper indicates that the call cannot proceed. If this occurs and the near-end gatekeeper is configured with the necessary route pattern, the next preferred trunk in the route pattern is used for that call as follows.

- Send LRQ.
- Wait 2sec for LCF (1sec as of Communication Manager 3.0).
- Send LRQ.
- Wait 2sec for LCF (1sec as of Communication Manager 3.0).
- Go to next preferred trunk in route pattern (4sec total per call for Communication Manager pre-3.0; 2sec total per call as of 3.0).

The LRQ feature affects individual calls, whereas the IP trunk bypass feature affects entire IP trunks. The IP trunk bypass feature takes some time to detect a problem in the IP network and put the signaling-group into bypass state. When this happens, with the appropriate route pattern in place, it results in all calls being routed onto the next preferred trunk. The LRQ feature speeds up per call re-routes until IP trunk bypass is established, so the two features can work in conjunction.
When LRQ is enabled the near-end listen port must be 1719. This means that the far-end gatekeeper must have its far-end listen port set to 1719. If the far-end gatekeeper is an Avaya call server and also has LRQ enabled (near-end listen port is 1719), then the near-end gatekeeper must have its far-end listen port set to 1719. Also, when LRQ is enabled calls cannot share the IP signaling connection, so this parameter must be set to 'n'. Each call establishes signaling across the IP trunk after a successful LRQ/LCF exchange.
For information about IP trunking with the Cisco Call Manager, see "Avaya S8300 Media Server and Avaya S8700 Media Server Networked with Cisco Call Manager using H.323 Signaling and IP Trunk Groups" at www.avaya.com.

media-gateway

Options are **add**, **change**, **display**, and **list**. This form is used to administer a G700/G350/G250 media gateway. Number is simply a numeric index. Type is the media gateway model (ie, G700, G350, G250, G250-BRI). Name is a text descriptor. Serial No is the gateway's serial number, which is displayed by typing **show system** at the MGP CLI. A gateway must be administered on the call server before it can register to that server, and the serial number is what uniquely identifies a valid gateway.
Network Region is used for IGAR purposes, similar to assigning port networks to a network region on the **cabinet** form. But unlike the **cabinet** form, the network region designation on the **media-gateway** form also assigns the gateway VoIP resources to a particular Communication Manager network region. This is equivalent to assigning a MedPro/MR320 board to a network region on the **ip-interfaces** form. Recovery Rule determines automatic recovery back to the primary server while the media gateway is registered to an LSP. The default is no automatic recovery ('none'), or a number can be placed here to apply a recovery rule, per the **system-parameters mg-recovery-rule** form, as explained in the following section.
Encrypt Link refers to the H.248 signaling link between the gateway and the call server.
Location serves the same function as the identical field on the **cabinet** form; it is used for call routing purposes (see the "Avaya Communication Manager Network Region Configuration Guide" at www.avaya.com). Site Data can be used to note the gateway's address (ie, if it is located at a remote branch office). For G250 models Max Survivable IP Ext refers to how many IP stations are permitted to fail over to the gateway when connectivity to the primary call server is lost. This is part of the SLS feature, new to Communication Manager 3.0 and the G250. The remaining information is automatically populated when the gateway registers with the call server.

system-parameters mg-recovery-rule

Options are **change** and **display**. When a media gateway loses connectivity to the primary call server, it can fail over to an LSP. This form, new to Communication Manager 3.0, administers rules that determine when a media gateway automatically recovers back to the primary server. The Number is simply a numeric index. Rule Name is a text descriptor. Migrate H.248 MG to primary and Minimum time of network stability are the two conditions that must be met before the primary Communication Manager server accepts a media gateway recovery registration.
First the minimum network stability time condition must be met. Then the recovery can happen…
- Immediately.
- When there are no active calls on the media gateway.
- During a specified time window.
- Either when there are no active calls, or during a specified time window.

A blank Migrate H.248 MG to primary field indicates that the rule is disabled. The failover to an LSP, and recovery back to the primary server, are covered in detail in the "Avaya Communication Manager Network Region Configuration Guide" at www.avaya.com.

system-parameters ip-options

Options are **change** and **display**. This form is used for miscellaneous IP settings.

IP Media Packet Performance Thresholds: These parameters, detailed in Appendix G, are for the IP trunk bypass feature described in the section covering the **signaling-group** form.

RTCP Monitor Server: These are the VoIP Monitoring Manager server settings applied to all network regions, unless specified otherwise in the **ip-network-region** form.

Automatic Trace Route on Link Failure: This feature relates to the following links.
- Port network control link between S87xx/S8500 server and IPSI board.

- H.248 media gateway control link between CLAN/S8300 and media gateway.
- IP trunk between near-end system and far-end system.

When this feature is enabled, and Communication Manager detects a failure on one of these links, Communication Manager launches a trace route from the source of the link to the destination of the link. A failed trace route might indicate that the link failure was associated with a network fault, whereas a successful trace route might indicate otherwise. This feature should be disabled if ICMP is blocked on the network, so as not to give false indications. The results of this trace route are logged on the call server, with an IPEVT tag (one of many events with that tag).

H.248 Media Gateway and H.323 IP Endpoint: See the "Avaya Communication Manager Network Region Configuration Guide" at www.avaya.com for information on most of the parameters under these headings. Only the Periodic Registration Timer is covered here. This timer determines the frequency at which a forcefully unregistered IP phone attempts to re-register. The primary application is for a desktop IP telephone that is forcefully unregistered because a user from home takes over the extension with a softphone. At some point the user logs off the softphone, leaving the extension free for the IP phone to reacquire. However, the IP telephone doesn't know when the softphone logs off, so the IP phone simply attempts to register periodically, and succeeds only after the softphone logs off. This timer determines that frequency, and it requires IP telephone 2.1 or later.

Music on Hold: This feature applies to media gateways and to port networks in IP-Connect systems with no traditional PNC (Center Stage or ATM). When music must be delivered via IP between media gateways and port networks, the music should be transported via the G.711 codec for quality reasons. If network region assignments are such that there is always a G.711 path between media gateways and port networks, this feature is not necessary. In some configurations there may not be a G.711 path, and in such cases setting this parameter to 'y' forces the use of G.711 for music transport.

IP DTMF Transmission Mode: The intra-system parameter determines how DTMF tones are passed within a system between media gateways and IP-connected port networks with no traditional PNC (Center Stage or ATM). The inter-system parameter, configured on the **signaling-group** form, determines how DTMF tones are passed between systems across IP trunks. Note that both ends of the IP trunk must be configured the same.
The primary issue driving these parameters is the fact that DTMF tones are not accurately reproduced using compressed codecs. This is particularly an issue for systems that rely on DTMF tones for functionality. The options operate as follows.
- in-band: If the configured codec is G.711 or G.729, the tones are passed in-band. Otherwise, the tones are passed out-of-band via call signaling. G.711 accurately passes DTMF tones, while G.729 can pass the tones but is susceptible to error. This option is obsolete on CM 3.1 for intra-system DTMF digits.
- in-band-g711: If the configured codec is G.711, the tones are passed in-band. Otherwise, the tones are passed out-of-band via call signaling. This option removes the uncertainty of G.729. This option is obsolete on CM 3.1 for intra-system DTMF digits.
- out-of-band: The digits represented by the tones are always passed out-of-band. If H.245 messages are exchanged, the H.245 UserInputIndication message is used to pass the digits. Otherwise, the Keypad Information Element of an H.225/Q.931 INFO message is used to pass the digits.
- rtp-payload: The digits represented by the tones are sent via the RTP payload format specified in RFC 2833. This is required by SIP but also applicable to H.323.

The last two options require the MedPro/MR320 board and VoIP media module to detect the tones and remove them from the outgoing audio stream. Then a message is sent to the call server for each digit to be sent out-of-band, or a separate RTP packet with the specified payload format is created for each digit.

SAT Troubleshooting Commands

The following table lists some common SAT troubleshooting commands.

status station <ext>	Gives static view of a station's status (multiple pages).
list trace station <ext>	Gives real-time view of a station's activities – for tracing calls.
list trace ras ip-stations <ext>	Traces a station's registration events (GRQ, GCF, RRQ, RCF).
status signaling-group <group #>	Gives status of signaling-group.
status trunk <group #> or **<group #>/<member #>**	Gives status of trunk-group or group/member.
status ip-board <slot #>	Gives Ethernet interface in/out statistics for an IP board.
status clan-port <slot #17> (ie, 01a0517)	Gives C-LAN board statistics (multiple pages).
status clan-usage	Gives C-LAN socket usage.
status media-processor all | board <slot #>	Gives MedPro/MR320 status for all boards or individual board.
status ip-network-region <#>	Gives status of inter-region connectivity.
list ethernet-options	Lists all administered speed/duplex settings for IP boards.
get ethernet-options <slot #>	Gives administered vs. actual speed/duplex settings for a board.
ping and **trace-route**	Sends pings and trace-route from a board or from a station. If board, specify **board <slot #>**. If station, specify **source <port #>**, where port # is from **status station** form. Use Help feature.

Table 7: Common SAT troubleshooting commands

4 Guidelines for Avaya 4600 Series IP Telephones

This section covers some general information regarding various Avaya 4600 Series IP Telephone models. More specific information is available in the "4600 Series IP Telephone LAN Administrator's Guide" and other IP telephone guides at support.avaya.com. The current GA firmware releases can be obtained at the same site. Be sure to read the "readme" files that accompany each firmware package.
Note: For simplicity in many IP telephone applications a C-LAN is often called a gatekeeper, although the call server is the gatekeeper and the C-LAN is only a front end to the gatekeeper.

4.1 Basics

Legacy Models vs. Current Models

Legacy Avaya IP Telephones such as the 4606, 4612, 4624, and 4630, contain a 10/100 hub. The last and best firmware for these models is version 1.8.3; legacy models cannot accept firmware newer than this.

The integrated hub in the legacy IP Telephones operate at 10 mbps, or 100 mbps half duplex. When connected to an Ethernet switch port that is configured to auto-negotiate, the Ethernet switch port stabilizes at 100/half. The exception to this is if a personal computer is attached to the telephone that is capable of only 10 mbps. In this case, all three devices stabilize at 10/half. If no personal computer is to be attached to the telephone, or if the attached computer will always be capable of 100 mbps operation, it is good practice to lock down the Ethernet switch to 100/ half. If a personal computer might be attached to the telephone, and there is a chance that the computer might have a 10-mbps NIC, leave the Ethernet switch port in auto-negotiate mode. These older telephones, however, cannot operate in full duplex mode.

The term "current models" in this document refers to the 4620 and 4610 and models containing the SW designation. Current model telephones have an internal Ethernet switch that allows the telephone and a PC to share the same LAN connection, if appropriate. Thus, SW models do not need, or work with, the 30A (applicable to the 4612/4624/4630 only) switched hub interface. The exception to this exception is the 4620--both the 4620 and 4620SW contain an Ethernet switch. The built-in 10/100 Ethernet switch permits speed and duplex configuration if necessary. This switch is set to auto-negotiate speed and duplex by default. The closest Ethernet switch to which the IP Telephone is attached should be set to auto-negotiate, as well. Locking down the closest switch to full duplex without also "locking down" the duplex of the phone will lead to packet loss, and thus result in problems with voice quality. Follow the guidelines in section 2.1, heading "Speed/Duplex" when configuring the speed and duplex on these phones and the Ethernet switch ports to which they are connected. Current models also have an updated look and a larger screen that facilitates additional features and functionality. Feature-related implementation, and the additional features and functionality of the current models are covered in IP telephone specific documentation found at support.avaya.com.

When the IP phone and PC are both transmitting, the phone's traffic is given strict priority out the uplink port to the enterprise Ethernet switch. This is not an issue for the PC, because under normal conditions the IP phone transmits less than 100kbps of audio traffic. Prioritization of traffic downstream from the enterprise Ethernet switch to the phone's switch port must be handled by the enterprise Ethernet switch.

The built-in Ethernet switch strips the 802.1Q tag from the IP telephone toward the PC. That is, tagged traffic from the phone is sent to the Ethernet switch (uplink port) with the tag, but to the attached PC (user port) without the tag. This also allows the attached PC to communicate with the IP telephone when they are on the same VLAN and the phone is tagging.

DHCP Option 176

Just the basics of DHCP option 176 are covered here. See the "4600 Series IP Telephone LAN Administrator's Guide" for more details.

The DHCP specification has what are called **options**, numbered from 0 through 255. Each option is associated with a specific bit of information to be sent by the DHCP server to the DHCP client. For example, option 1 is the subnet mask option and is used to send the subnet mask to the client. Option 3 is the router option and is used to send the default gateway address and other gateway addresses to the client. Some options are defined – such as options 1 and 3 – and others are not. The defined options are found in RFC 2132.

Options 128 through 254 are **site-specific options**. They are standard options that are not defined, and vendors may use these options and define them to be whatever is necessary for a specific application. Avaya IP telephones use site-specific option 176 as one of the methods to receive certain parameters from the DHCP server.

For the Avaya application of option 176, it is defined as a string. The string contains parameters and values separated by commas, as illustrated after the following table. The most prevalent parameters and values are as follows.

Parameter	Value
MCIPADD	Address(es) of gatekeeper(s) – at least one required
MCPORT	The UDP port used for registration – 1719 default
TFTPSRVR TLSSRVR HTTPSRVR	Address(es) of TLS/HTTP(S)/TFTP server(s) – at least one required
L2QVLAN	802.1Q VLAN ID – 0 default
L2QAUD	L2 audio priority value.
L2QSIG	L2 signaling priority value.
VLANTEST	The number of seconds a phone will attempt to return to the previously known voice VLAN

Table 8: DHCP option 176 parameters and values

The typical option 176 string for a single-VLAN environment looks like this.

MCIPADD=*addr1*,*addr2*,*addr3*, … ,MCPORT=1719,HTTPSRVR=*addr*

At least one gatekeeper (C-LAN or S8300 or S8500 Main Server) address must be present after MCIPADD to point the phones to a call server. MCPORT specifies which UDP port to use for RAS registration. IP telephone firmware 1.6.1 and later already have 1719 as the default port, but it is prudent to include it. A TFTP server address is necessary so that phones know where to go to download the necessary script files and binary codes (see "Boot-up Sequence" heading below). **L2QVLAN** and **VLANTEST** would be included if 802.1Q tagging were required, such as in a dual-VLAN environment (see section 4.2). Other parameters may be added, such as **L2QAUD** and **L2QSIG**, which are used to specify the L2 priority values for audio and signaling. If these values are not specified in option 176, the default values (6/6) are used.

Note: The L3 priority values (DSCP) are received from the call server, as configured on the SAT **ip-network-region** form. The reason L3 values are received from the call server and L2 values are not is because an IP phone accepts all L2 values from one source. The preferred and recommended method is via DHCP option 176. An alternative method is described in section 3.5, heading "ip-network-map," which utilizes the L2 values administered on the SAT **ip-network-region** form.

An administrator must create option 176 on the DHCP server and administer a properly formatted string with the appropriate values. Option 176 could be applied globally or on a per scope basis. The recommendation is to configure option 176 on a per scope basis, because the values themselves or the order of the values could change on a per scope basis. As part of the DHCP process at boot-up, the IP telephone requests option 176 from the DHCP server.

DHCP Lease Duration

A DHCP server gives out an IP address with a finite or infinite lease, and the Avaya recommended lease duration for IP phones is 2 to 4 weeks. The DHCP specification calls for the client to renew the lease at determined intervals, typically beginning at half-life of the lease. If the first renewal attempt fails, there are allowances in the specification for further renewal attempts, dependent on the length of the lease. Too short a lease requires too many renewals, which not only taxes the DHCP server but can also disrupt service to the IP phones if renewals cannot be accomplished for whatever reason. On the other hand, too long a lease can result in IP address exhaustion if hosts are unplugged from the network without properly shutting them down to invoke a release of the IP address lease.

Additional Script and Firmware Download Methods

Beginning with Avaya IP Telephone Release 2.2, Avaya IP phones can download scripts and firmware from web servers using the HTTP or TLS (HTTPS) protocols, in addition to TFTP. Preliminary testing at Avaya Labs indicates that HTTP servers can support more simultaneous downloads than TFTP servers, suggesting that HTTP/TLS are better suited for large IP telephone deployments than TFTP.
To specify TLS or HTTP script/firmware downloads, in option 176 of the DHCP scope apply the **TLSSRVR** (for TLS) or **HTTPSRVR** (for HTTP) parameter in lieu of **TFTPSRVR**. If TLSSRVR, HTTPSRVR, and TFTPSRVR are all set, the phone will attempt to download firmware using TLS first on TCP port 411, then HTTP on TCP port 81, then HTTP on TCP port 80, then TFTP on UDP port 69.
Note: Avaya IP telephones only establish encrypted TLS connections with servers using an Avaya-signed digital certificate (ie, an Avaya S8300 or S8500 Media Server).

Boot-up Sequence

The following are key boot-up events, listed in order, which may help to verify proper operation of the IP phone. This list may not be comprehensive, as only key events are listed. The packets described here can be captured using a protocol analyzer, and one with H.323 capability is required to properly decode the H.225 RAS messages. On 4606/12/24 models the analyzer can be attached to the phone's user port. But because the 4620 and 4610 have a built-in switch instead of a hub, the analyzer must be attached to a mirrored Ethernet switch port, or to a tap or hub in line between the phone and the Ethernet switch.

- Initial startup – At power-up or manual reset, the phone goes through a short initial startup procedure. The display shows `Restarting…` (if the phone was intentionally restarted w/ **Hold RESET#**), and then `Loading…` and `Starting…`
- DHCP – The phone queries the DHCP server for an IP address and other needed information. The following packets are exchanged: DHCP Discover from phone to broadcast; DHCP Offer from server to broadcast, or relay agent to phone; DHCP Request from phone to broadcast; and DHCP ACK from server to broadcast, or relay agent to phone. *Note that this step is bypassed if the phone is manually configured with all the necessary information.*
- Request file "46xxupgrade.scr" and others from TLS/HTTP/TFTP server – This is a text script file that tells the phone which boot code and application code are needed. If the phone does not have the current codes, it requests them from the file server. A brand new phone makes all three requests, as phones typically come from the factory with outdated code. In addition, the "46xxupgrade.scr" script may instruct the phone to download the "46xxsettings.scr (or .txt)" file, which is an optional method of sending configurations to the phone. *Note that there is a loading period after each .bin code is received for the first time. Note also that the file names are case sensitive on some servers (Unix/Linux) and not on others (Microsoft).*
- Ext and Password prompts – The phone prompts for the extension and password if there are no previously stored values.
- Registration with gatekeeper – The phone registers with a gatekeeper (C-LAN or S8300) after the extension and password are entered. This registration happens very quickly and does not show up on

the display. However, the following packets are exchanged: RAS-Gatekeeper Request (GRQ) from phone to gatekeeper; RAS-Gatekeeper Confirm (GCF) from gatekeeper to phone; RAS-Registration Request (RRQ) from phone to gatekeeper (not necessarily the same one GRQ was sent to); RAS-Registration Confirm (RCF) from gatekeeper to phone.

- H.225 call signaling connection – The phone opens a TCP session with the gatekeeper and sends an H.225 Setup message, which is answered with H.225 Call Proceeding and H.225 Connect messages from the gatekeeper. This call signaling session remains up throughout the registration. During idle periods the phone maintains the session by sending TCP keepalives.
- Phone is operational – The administered display shows up on the phone (and the extension LED illuminates on 4606/12/24 models).
- Unregistration messages – If the gatekeeper intentionally unregisters a set, or if the set intentionally unregisters itself, the message sent by either the gatekeeper or the set is a RAS-Unregistration Request (URQ) with a reason code that is deciphered in the hex decode of most protocol analyzers. The acknowledgment message is a RAS-Unregistration Confirm (UCF).

Call Sequence

It is not feasible to give a standard packet-by-packet call sequence, because of the many possible variations on any given call. Instead, a higher level description of the process is offered here. Depending on which features are enabled and executed during a call the packet-by-packet sequence may vary, but the fundamental functions described here apply overall. All call signaling functions go through the gatekeeper, either via the C-LAN or natively (S8300), and the gatekeeper dictates what the IP stations do during a call.

- Calling phone contacts gatekeeper on already established call signaling session (TCP 1720 gatekeeper port, variable phone port).
- There are some call signaling exchanges on this TCP session.
- Calling phone establishes an audio stream with an audio resource (MedPro/MR320 board or VoIP module), as directed by the gatekeeper.
- Gatekeeper contacts called phone on already established call signaling session (TCP 1720 gatekeeper port, variable phone port).
- There are some call signaling exchanges on this TCP session.
- Called phone also establishes an audio stream with an audio resource, as directed by the gatekeeper, but this stream is one-way until the call completes.
- Called phone answers, resulting in more call signaling activity, and the call completes. The call could remain in this state, but…
- In most cases, unless configured otherwise, the gatekeeper contacts both phones and instructs them to direct their audio streams to each other.
- Phones direct audio streams to each other, as instructed by the gatekeeper.
- One of the phones hangs up, resulting in more call signaling activity.
- Gatekeeper contacts both phones, signals that the call has ended, and instructs them to tear down audio streams.
- Phones tear down audio streams.

Keepalive Mechanisms

There are two types of keepalive mechanisms: RAS and TCP.

- RAS keepalive – The IP telephone sends RAS keepalive messages to the gatekeeper at a time-to-live (TTL) interval specified by the gatekeeper. On a protocol analyzer a RAS keepalive message shows up as a RAS-Registration Request (RRQ) with the keepalive bit set in the RAS decode. Each request message is acknowledged by the gatekeeper with a RAS-Registration Confirm (RCF). This exchange takes place over the RAS socket, which has UDP port 1719 on the gatekeeper side.

- TCP keepalive – The IP telephone sends TCP keepalive messages to the gatekeeper at a regular interval determined by the phone, or as administered on the **ip-network-region** form. The keepalive is an empty TCP datagram with a sequence number that is 1 to 5 less than the sequence number of the previous real TCP message or ACK sent by the phone. The gatekeeper acknowledges each keepalive from the phone with a similar empty TCP datagram. This exchange takes place over the call signaling socket, which has TCP port 1720 on the gatekeeper side. The CLAN sends TCP keepalives, similar to the phone's TCP keepalives. However, because a CLAN must keep track of potentially hundreds of phones, the CLAN's keepalive intervals are much longer than the phone's keepalive intervals. A CLAN sends regular keepalives to every phone once every 10min. These keepalives are not synchronized, so they don't all go out to every phone at the same time. If one of the 10-min keepalives is missed, the CLAN sends five more retry keepalives 1min apart. So a link bounce detection time for a CLAN is 5-15min. If a phone becomes unreachable and does not re-register for an extended period of time, it takes the CLAN 5-15min to discover that the phone is no longer reachable. This means it takes CM 5-15min to internally unregister that phone. The detection time is much faster – less than a minute – if CM tries to deliver a call to that phone and fails.
- Regular and retry intervals – Each keepalive mechanism has a **regular interval** as described above. If a regular interval keepalive is not acknowledged, more keepalives are sent at a faster **retry interval**. If all the retry keepalives are unanswered, the phone effectively unregisters and moves on to the next gatekeeper in its gatekeeper list (obtained via DHCP and/or the gatekeeper).
- TTL – As stated above, the gatekeeper sends a TTL for the RAS keepalive mechanism. The TTL is the greater of 60 seconds or a multiplier times the number of registered endpoints. The multiplier for a CM server is approximately 1.4 seconds, which means that anything above 42 registered endpoints would exceed the minimum 60-sec TTL. The multiplier for the other servers described in this document is .1 second, which means that more than 600 registered endpoints are required to exceed the minimum 60-sec TTL.

Independent of the mechanism (RAS or TCP), the keepalive flow follows this pattern.

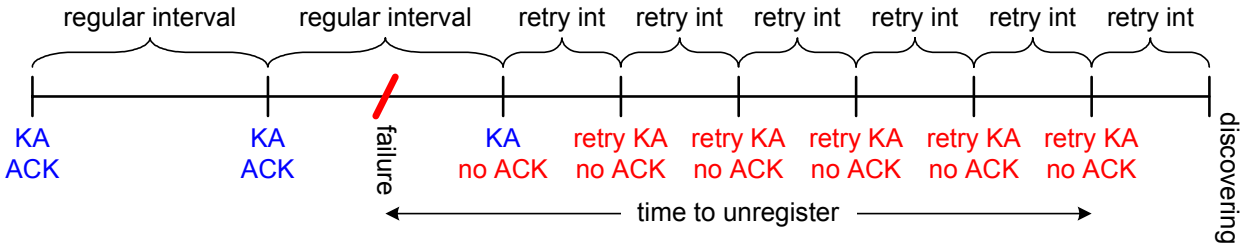


Figure 17: Keepalive pattern

The **discovering** at the end of the flow means that the phone has effectively unregistered and is searching for another gatekeeper. Effectively unregistered means that the phone has not sent an explicit RAS-Unregistration Request (URQ) message, but it considers itself unregistered from that gatekeeper and is moving on to the next. Even if the phone did send a URQ, chances are the gatekeeper would not receive it because the failure condition could still exist.

The final retry interval prior to discovering would appear to give extra time for the failure to recover. And indeed if the phone did receive a KA acknowledgment within that final retry interval it would stay registered to the same gatekeeper. However, the reality is that if the phone doesn't receive an acknowledgment within a second or two after the final retry KA, it won't receive one. Therefore, the final retry interval really does not factor into the **time to unregister**. Time to unregister answers the question, "How long must the failure (ie, network outage) last before the IP telephone unregisters?" If the failure recovers just before the final retry KA is sent, the phone remains registered to the same gatekeeper. If the failure recovers a couple seconds after the final retry KA is sent, the phone most likely unregisters and moves on to the next gatekeeper after the final retry interval.

The TCP and RAS keepalive algorithms are as follows.

IP telephone	TCP KA regular intrvl	TCP KA retry intrvl	Time to unregister	RAS KA regular intrvl	RAS KA retry intrvl	Time to unregister
4620/10 2.x and later w/ CM2.x and later	20sec configurable	5 * 5sec configurable	25 to 45sec varies	obsolete	Obsolete	n/a

Table 9: TCP and RAS keepalive matrix

4.2 Connecting a PC to the Phone

On the back of the phone, the port with the icon that looks like a terminal is the user port. (The port with the icon that looks like a network jack is the uplink port, which connects to the Ethernet switch.) Use discretion when connecting a PC to the phone, and remember that its primary function is not that of an enterprise network device. For example, do not connect an enterprise server to the phone. Such high-traffic servers require their own separate connections to the enterprise Ethernet switch. Also, do not connect a PC to the phone with a 10M uplink to the network. The phone itself operates well at 10M, but with a PC attached the two should operate at 100M.

IP Phone and Attached PC on Same VLAN

There are three variations of attaching a PC to the phone, and the first two involve having both the phone and the PC on the same VLAN, which is the port/native VLAN (*refer to Appendix A for a primer on VLANs*). In the first scenario, traffic from both the phone and the PC have no CoS tagging. In this case, no special configurations are necessary. Simply attach the phone to an access port (one with only the port/native VLAN configured) and attach the PC to the phone.
The second scenario is similar to the first, except that traffic from the phone is marked with L2 and/or L3 priority while remaining on the port/native VLAN. See the instructions in section 2.3 under the heading "Rules for 802.1p/Q Tagging." The phone must be configured to apply the appropriate L2 and/or L3 priority values. The **Hold ADDR#** menu is used to manually enable or disable 802.1Q tagging and to set the VLAN ID. The other parameters are configured via the **Hold QOS#** menu. The manual method is covered below, and an automated method is covered in the next paragraph.

- 802.1Q – On/off for 802.1Q tagging. Turn this on if L2 priority tagging is desired; off otherwise.
- VLAN ID – Should be zero (0) for this scenario, per the instructions in section 2.3, heading "Rules for 802.1p/Q tagging." The VID has no effect when 802.1Q tagging is disabled.
- VLANTEST – Not relevant when VID is zero. Applies in a dual-VLAN environment when VID is a non-zero value, as explained in later sections.
- L2 audio – Layer 2 CoS tag for Ethernet frames containing audio packets. The phone either receives this from DHCP (most common) or from the call server (rare), per the **ip-network-region** form. This value could also be set manually on a per phone basis.
- L2 signaling – Layer 2 CoS tag for Ethernet frames containing signaling packets. The phone either receives this from DHCP (most common) or from the call server (rare), per the **ip-network-region** form. This value could also be set manually on a per phone basis.
- L3 audio – Layer 3 DSCP for audio IP packets. The phone automatically receives this value from the call server, per the **ip-network-region** form. This value could also be set manually on a per phone basis.
- L3 signaling – Layer 3 DSCP for signaling IP packets. The phone automatically receives this value from the call server, per the **ip-network-region** form. This value could also be set manually on a per phone basis.

The manual menus are covered here for explanatory purposes. However, a better alternative is to use DHCP option 176 and the built-in capabilities of the call server and IP telephone to automatically configure the phones. As stated previously, the call server sends the L3 priority values to the phones

automatically, per the values configured in the **ip-network-region** form. The 802.1Q on/off instruction, VLAN ID, and L2 priorities can be configured automatically using DHCP option 176 as described in section 4.1, heading "DHCP Option 176." Here is what that string should look like for 1.8 and later phones (see the appropriate "LAN Administrator's Guide" for previous phone releases).

MCIPADD=*addr1*,*addr2*, … , ,HTTPSRVR=*addr*,L2QVLAN=0,L2QAUD=#,L2QSIG=#

The **L2QVLAN=0** parameter instructs the phone to enable 802.1p/Q tagging with VID 0, which means that the phone's traffic belongs on the port/native VLAN. The Ethernet switch port to which the phone is connected must be configured to accept 802.1Q tagging for this to work, and the switch must interpret VID 0 as the port/native VID, per the IEEE 802.1Q standard [6 p.69]. If the Ethernet switch does not understand VID 0, the phone may need to tag with the port/native VID, although this is not the standard method.

Remember that in order for the CoS markings to have any effect, the corresponding QoS configurations must be implemented on the necessary network devices. Remember also that improperly enabling L2 and L3 prioritization may break processes that were working without it. Read section 2.3 of this document for more information on CoS and QoS.

IP Phone and Attached PC on Different VLANs

The third and most common scenario for attaching a PC to the phone (the first two were covered in the previous heading) is to have the phone and PC on separate VLANs. This requires a dual-VLAN port on the Ethernet switch as described in section 2.3, heading "Rules for 802.1p/Q Tagging." One of the VLANs is the port/native VLAN (the data VLAN), and clear Ethernet frames (ones with no 802.1Q tag) from the PC are forwarded on this VLAN. The other VLAN is the voice VLAN, and the IP phone must tag its traffic with the proper VLAN ID to have it forwarded on this VLAN.

The **Hold ADDR#** and **Hold QOS#** menu options are the same as described in the previous heading, except that the VID must not be zero. The preferred method of using DHCP option 176 (section 4.1, heading "DHCP Option 176") is also the same, except that **L2QVLAN** has a non-zero value. Finally, in a dual-VLAN implementation the **VLANTEST** parameter has great significance, as illustrated below. The following scenario, with arbitrary voice VLAN ID, details the steps a phone (1.8 and later) would go through in a typical dual-VLAN implementation. It also illustrates the recommended content of the option 176 string.

- Phone with no previously stored values boots up and obtains an address on the data VLAN.
- The data VLAN option 176 string directs the phone to go to voice VLAN 25.

MCIPADD=*addr1*,*addr2*, … , HTTPSRVR=*addr*,L2QVLAN=25,L2QAUD=6,L2QSIG=6,VLANTEST=600

Phone releases the data VLAN address and obtains an address on the voice VLAN.

- The voice VLAN option 176 string is identical to the data VLAN string but without the **L2QVLAN** parameter, because a phone already on the voice VLAN doesn't need to be directed to go there.

  MCIPADD=*addr1*,*addr2*, … ,MCPORT=1719,TFTPSRVR=*addr*,L2QAUD=6,L2QSIG=6,VLANTEST=600
- Phone is operational on the voice VLAN.
- Reboot or power cycle occurs.
- Phone immediately returns to voice VLAN 25 upon recovery, and one of the following occurs.
  - Phone obtains an address and option 176 string on the voice VLAN and all is well.
  - Phone cannot obtain an address on the voice VLAN, due to network or DHCP problems. In this case the **VLANTEST=600** parameter directs the phone to continue trying for 600sec (finite range is 1-999). If the phone does not succeed in obtaining an address within 600sec, it marks VLAN 25 as invalid and returns to no tagging (back to the data VLAN).

The idea behind going back to the data VLAN after some time is that the phone may have changed ports and be on one with a different voice VLAN. In such a case the phone would have to start over and be directed to the proper voice VLAN. The idea behind marking VLAN 25 as invalid in the previous scenario is that if the phone hasn't changed ports, it is preferable to operate on the data VLAN than to be

sent to a bad voice VLAN in a continuous loop. For cases where it is not preferable to operate on the data VLAN, the option **VLANTEST=0** was added as of legacy phone firmware 1.8.2 and current phone firmware 2.0.1. This instructs the phone to permanently remain on the previously known voice VLAN. Once the phone accepts VLANTEST=0 or marks a VLAN as invalid, the only way to clear out this state is by manually resetting the values via the **Hold RESET#** menu.

Note: DHCP option 176 is the preferred method for directing IP phones to the voice VLAN. The method described previously using the VLAN field of the **ip-network-map** form is an alternative if DHCP option 176 is not available. The two methods should not be used simultaneously.

Phone firmware 2.4.1 changes the values and behavior of VLANTEST. The value range increases from 0 - 999 (16.65 minutes) seconds to 0 - 172800 (48 hours). Also, when the timer expires, the voice VLAN is NOT marked as invalid. Instead the DHCP requests shift to the data VLAN for the value of VLANTEST seconds. If it is not answered it will shift back to the voice vlan for the same value of VLANTEST seconds. DHCP requests will continually alternate between data and voice VLANs. You no longer have to manually reset values to clear the information from memory. Using VLANTEST=0 still works the same. It will disable moving from the voice VLAN back to the data VLAN by keeping all requests on the voice VLAN.

Appendix A describes how to configure a simple network for dual-VLAN operation.

Remember that in order for the CoS markings to have any effect, the corresponding QoS configurations must be implemented on the necessary network devices. Remember also that improperly enabling L2 and L3 prioritization may break processes that were working without it. Read section 2.3 of this document for more information on CoS and QoS.

4.3 Gatekeeper Lists and DHCP Option 176

An IP telephone can have a list of gatekeepers (C-LANs and, S8300s, or S8500 Main Servers) to which it may send the initial RAS-Gatekeeper Request (GRQ) message. This list is obtained via the DHCP option 176 string, which is covered briefly in section 4.1 and in detail in the "LAN Administrator's Guide." Within the DHCP option 176 string, the comma-separated IP addresses that follow the **MCIPADD** parameter constitute a gatekeeper list, and this list provides redundancy at boot-up. If a given gatekeeper is unreachable for any reason, the phone attempts other gatekeepers in the gatekeeper list. The following hypothetical network diagram and the accompanying instructions explain how gatekeeper lists should be administered on DHCP servers.

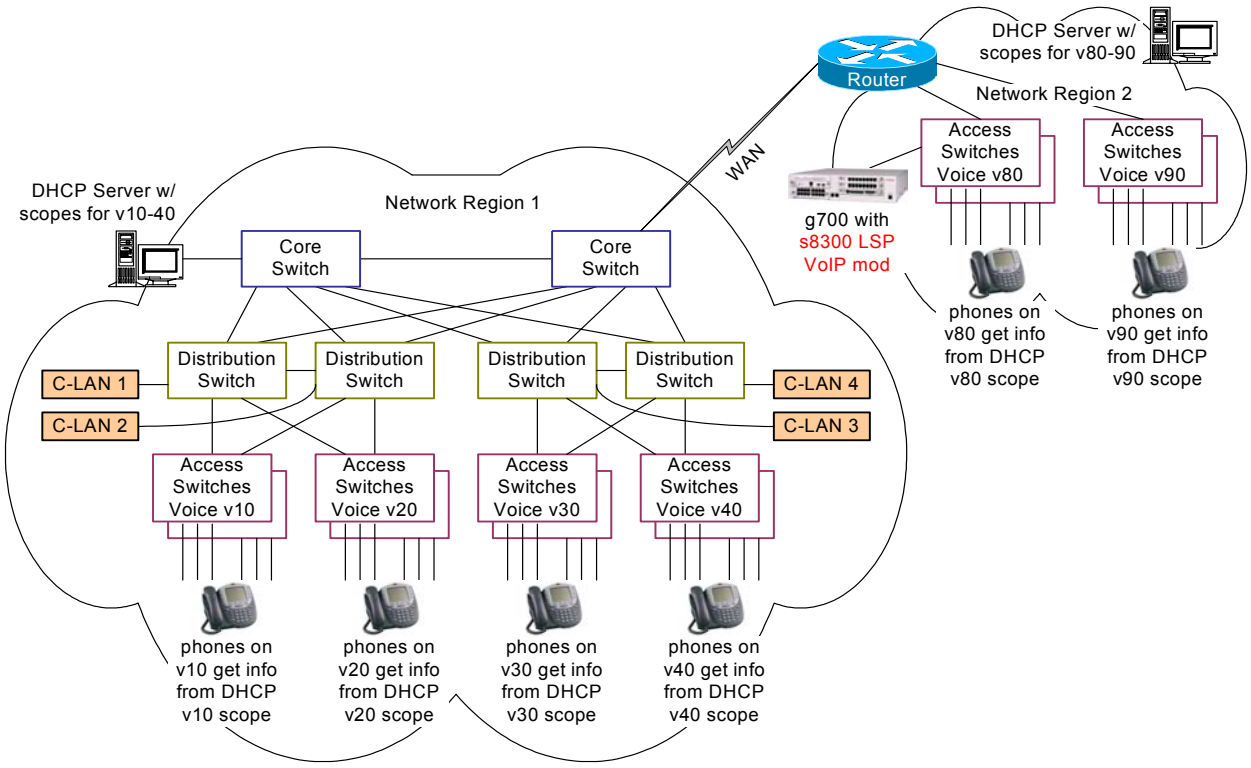


Figure 18: Hypothetical converged network

Main Site

The converged network depicted in the figure above could be an entire network, or a portion of a much larger network. The main site is implemented in a core-distribution-access architecture common to many enterprise networks. The IP phones are scattered across various voice VLANs, but the phones all belong to the same Communication Manager network region because they use the same codec set, share the same audio characteristics, and use the same resources specified by a network region. Network region 1 has four C-LANs scattered across four distribution switches, but there could be more depending on the number of IP telephones. The fact that there are four C-LANs and four voice VLANs is purely coincidental.

Suppose, for whatever reason, that a large number of IP phones are rebooted at once. Which gatekeeper(s) will they contact first? The correct answer is that they should contact all the gatekeepers in a distributed fashion. All the phones should not bombard the same gatekeeper at once with GRQs. There are various ways to configure the gatekeeper lists, and the following is possibly the simplest.

- v10 scope: "MCIPADD=*clan**1**addr,clan**2**addr,clan**3**addr,clan**4**addr*, …"
- v20 scope: "MCIPADD=*clan**2**addr,clan**3**addr,clan**4**addr,clan**1**addr*, …"
- v30 scope: "MCIPADD=*clan**3**addr,clan**4**addr,clan**1**addr,clan**2**addr*, …"
- v40 scope: "MCIPADD=*clan**4**addr,clan**1**addr,clan**2**addr,clan**3**addr*, …"

Based on how this particular network is implemented, here is another alternative.

- v10 scope: "MCIPADD=*clan**1**addr,clan**2**addr,clan**3**addr,clan**4**addr*, …"
- v20 scope: "MCIPADD=*clan**2**addr,clan**1**addr,clan**4**addr,clan**3**addr*, …"
- v30 scope: "MCIPADD=*clan**3**addr,clan**4**addr,clan**1**addr,clan**2**addr*, …"
- v40 scope: "MCIPADD=*clan**4**addr,clan**3**addr,clan**2**addr,clan**1**addr*, …"

Regardless of how the lists are administered, the principle is important. DHCP scopes should have rotating/varying gatekeeper lists, so as to produce a uniform distribution of GRQs at boot-up. Most DHCP servers facilitate this by permitting the option 176 string to be created per scope, which is

recommended. Do not create a global option 176 string that would apply to every scope on a server, resulting in only one gatekeeper list.
Note that this principle may also apply to multiple TFTP servers.

Branch Site

The branch site is just slightly different in terms of the DHCP scopes, but very different in terms of the failure scenario and other factors that affect the branch implementation.
The IP telephones at the branch site could access the same four C-LANs shown above, or there could be a different set of C-LANs (not shown) for the branch IP phones. In either case the DHCP scopes for v80 and v90 should have rotating lists, as at the main site. However, in addition to the list of C-LAN addresses, the v80 and v90 scopes should also include the S8300 LSP address at the end of the list. This is because the LSP can take over as the call server for the branch if the WAN link fails. The LSP only accepts registrations when it is active, so having the LSP in the list does not result in inadvertent registrations to the LSP.
Because an extended WAN link failure is possible, the branch site should ideally have its own DHCP server. It makes sense that if there is a redundant call server at the branch, there should also be a dedicated DHCP server, because IP telephones require both services. For cost and administrative reasons, however, many will choose not to install a DHCP server at all branch locations. In such cases it is very important that the IP telephones not be rebooted during a WAN link failure, because they would not be able to obtain IP addresses. The manual configuration option is available, but it is not always a viable option for various reasons.

Two Methods of Receiving the Gatekeeper List

In addition to receiving the gatekeeper list via DHCP option 176, as described above, a gatekeeper list is also received via the RCF message during registration. In other words, when an IP telephone registers with the call server, the call server sends a gatekeeper list in the RCF message. The H.323 standard calls this the **Alternate Gatekeeper List**. This means that a phone really only needs one gatekeeper address at boot-up, because the phone receives the gatekeeper list when it registers. This feature is useful for phones that are manually administered, as the manual method only permits the entry of one gatekeeper address. However, it is still preferable to administer a gatekeeper list in DHCP option 176 for redundancy during boot-up.
Here are some key points regarding the option 176 gatekeeper list and the RCF Alternate Gatekeeper List.

- IP telephone versions prior to 2.0 use both lists simultaneously. GK addresses received from either method are merged into one list.
- IP telephone 2.0 and later maintain the two lists separately, with only one list active at any given time. During boot-up the phone uses the list obtained from option 176. After registration the phone uses the Alternate Gatekeeper List received in the RCF. When a phone is logged off but not rebooted, it reverts back to the list obtained from option 176.
- The option 176 GK list is recommended, as opposed to manual entry or a single GK address in option 176, because the RCF list is received after registration. If the phone only knows of one GK at boot-up and that GK is out of service, the phone cannot register and hence cannot get an RCF.
- The Alternate Gatekeeper List sent in the RCF follows a specific algorithm. When an IP phone registers and its network region is specified in the **ip-network-map** form, the call server delivers a list of all gatekeepers in that region, plus directly connected regions (specified in the **ip-network-region** form). If an IP phone's network region is not administered in the **ip-network-map** form, it inherits the region of the gatekeeper that receives the registration, and the call server delivers a list of all gatekeepers only in that region. The ip-interface form includes an administered Gatekeeper Priority Value between 1 and 9 where 1 is the highest priority. This value is used to build the RCF Alternate Gatekeeper list delivered in the RCF message. If multiple gatekeepers have the same priority value then the gatekeeper list is based on socket load per gatekeeper within the same priority.

- As of Avaya Communication Manager 1.3, the addresses of the LSPs (administered on the **ip-network-region** form) in the same network region as the IP phone are also sent in the RCF. As of Communication Manager 2.0, in addition to the LSPs, the address of the Survivable GK Node Name (administered on the **station** form) is also sent in the RCF.
- The combination of Communication Manager 2.x and IP telephone 2.x facilitates a distinction between primary and secondary gatekeepers in the Alternate Gatekeeper List. During recovery after an outage, the primary gatekeepers are attempted first for a period of time called the H.323 Primary Search Time, specified in the **system-parameters ip-options** form. After this search time expires, the secondary gatekeepers – LSPs and the Survivable GK – are also included in the search. For a more detailed discussion see the H.323 Link Bounce section of the "Avaya Communication Manager Network Region Configuration Guide" at www.avaya.com.

Verifying the Gatekeeper Lists

The table below gives a summary of how to view the gatekeeper and gatekeeper list in use.

Phone state **Method**	Registered phone	Phone logged off via **Hold LOGOFF#** keypad command
MIB Object ID .1.3.6.1.4.1.6889.2.69.1.1.3 (endptMCIPADD)	- 2.1 and later shows Alternate Gatekeeper List received in RCF message, which is the list in use. - 2.0.1 shows gatekeeper list received from option 176, or manually configured gatekeeper, even though the list in use is the Alternate Gatekeeper List from RCF. - 1.8.x shows combined list from RCF and option 176, or combined RCF list and manually configured gatekeeper.	- 2.1 and later shows gatekeeper list received from option 176, or manually configured gatekeeper. - 2.0.1 shows gatekeeper list received from option 176, or manually configured gatekeeper. - 1.8.x shows combined list from RCF and option 176, or combined RCF list and manually configured gatekeeper.
MIB Object ID .1.3.6.1.4.1.6889.2.69.1.1.4 (endptMCIPINUSE)	Shows gatekeeper to which phone is currently registered.	Shows gatekeeper to which phone was last registered.
MIB Object ID .1.3.6.1.4.1.6889.2.69.1.1.4.28 (endptRASGkList)	2.2 and later shows the alternate gatekeeper list received from CM in the RCF	2.2 and later shows the alternate gatekeeper list received from CM in the RCF
Hold ADDR# keypad menu	Shows gatekeeper to which phone is currently registered.	N/A

Appendix A: VLAN Primer

This appendix is primarily concerned with configurations that require the Avaya IP Telephone to connect to an Ethernet switch (Eth-switch) port configured with multiple VLANs – the IP phone on one VLAN and a PC connected to the phone on a separate VLAN. Three sets of configurations are given: Avaya P330 v3.2.8 and later, Cisco CatOS, and some Cisco IOS.

VLAN Defined

With simple Eth-switches, the entire switch is one L2 broadcast domain that typically contains one IP subnet (L3 broadcast domain). Think of a single VLAN (on a VLAN-capable Eth-switch) as being equivalent to a simple Eth-switch. A VLAN is a logical L2 broadcast domain that typically contains one IP subnet. Therefore, multiple VLANs are logically separated subnets – analogous to multiple switches being physically separated subnets. A L3 routing process is required to route between VLANs, just as one is required to route between switches. This routing process can take place on a connected router or a router module within a L2/L3 Eth-switch. If there is no routing process associated with a VLAN, devices on that VLAN can only communicate with other devices on the same VLAN.

For a tutorial and more information on VLANs, see "LANs and VLANs: A Simplified Tutorial" at www.avaya.com.

The *Port* or *Native* VLAN

Port VLAN and *native VLAN* are synonymous terms. The IEEE standard and most Avaya switches use the term *port VLAN* [6 p.11], but Cisco switches use the term *native* VLAN. Issue the command **show trunk** on Avaya P330/C360 and Cisco CatOS switches to see which term is used in the display output.

Every port has a port/native VLAN. Unless otherwise configured, it is VLAN 1 by default. It can be configured on a per port basis with the following commands.

Avaya P330 and C360	Cisco CatOS
set port vlan <id> <mod/port>	**set vlan <id> <mod/port>**

All clear Ethernet frames (ones with no 802.1Q tag, such as from a PC) are forwarded on the port/native VLAN. This is true even if the Eth-switch port is configured as an 802.1Q trunk, or otherwise configured for multiple VLANs (see VLAN binding heading below).

Configuring a Trunk

A trunk port on an Eth-switch is one that is capable of forwarding Ethernet frames on multiple VLANs via the mechanism of VLAN tagging. IEEE 802.1Q specifies the standard method for VLAN tagging. Cisco also uses a proprietary method called ISL. Avaya products do not interoperate with ISL.

A trunk link is a connection between two devices across trunk ports. This can be between a router and a switch, between two switches, or between a switch and an IP phone. Some form of trunking or forwarding multiple VLANs must be enabled to permit the IP phone and the attached PC to be on separate VLANs. The following commands enable trunking.

Avaya P330 and C360	Cisco CatOS
set trunk <mod/port> dot1q By default only the port/native VLAN is enabled on the trunk port. Another set of commands is required to specify other allowed VLANs.	**set trunk <mod/port> nonegotiate dot1q** By default all VLANs (1-1005) are enabled on the trunk port. VLANs can be selectively removed with the command **clear trunk <mod/port> <vid>**.

Note that Avaya adds additional VLANs to a trunk port that has only one VLAN, while Cisco removes excess VLANs from a trunk port that has all VLANs. Either method achieves the desired objective,

which is to have only two VLANs configured on a trunk port connected to an IP phone, so that broadcasts from non-essential VLANs are not permitted to bog down the link to the IP phone.

VLAN Binding Feature (P330/C360)

On the Avaya P330/C360, additional VLANs are added to a port using the VLAN binding feature. The port may be a trunk port (802.1Q tagging enabled) or an access port (no 802.1Q tagging). The port does not need to be a trunk to forward multiple VLANs, and for one application – connecting to an Avaya IP phone – it must not be a trunk (ie, do not issue the **set trunk** command). The following steps enable VLAN binding.

1. Verify that the port is configured with the desired port/native VLAN.
2. Add additional VLANs with one of the following **vlan-binding-mode** options.

Static option:	
set port vlan-binding-mode <mod/port> static	Put the port in bind-to-static mode.
set port static-vlan <mod/port> <vid>	Statically add another VLAN, in addition to the port/native VLAN.

----- OR -----

Configured option:	
set vlan <id>	Add a VLAN to the *configured* VLAN list. Type **show vlan** to see entire list.
set port vlan-binding-mode <mod/port> bind-to-configured	Apply the configured VLANs to the port and permit only those VLANs (**bind-to-all** permits all VLANs and not just the configured).

3. If the port is connected to a router or to another switch, trunking must be enabled with the command **set trunk <mod/port> dot1q**, which causes all egress frames to be tagged. However, if the port is connected to an Avaya IP phone with an attached PC, trunking must not be enabled so that none of the egress frames are tagged. This is necessary because most PCs do not understand tagged frames.

Setting the Priority without Trunking or VLAN binding (Single-VLAN Scenario)

With Avaya switches it is possible to set the L2 priority on the IP phone, even if the phone is not connected to a trunk or multi-VLAN port. That is, the Avaya switch does not need to be explicitly configured to accept priority-tagged Ethernet frames on a port with only the port/native VLAN configured. This is useful if the phone and the attached PC are on the same VLAN (same IP subnet), but the phone traffic requires higher priority. Simply enable 802.1Q tagging on the IP phone, set the priorities as desired, and set the VID to zero (0). Per the IEEE standard, a VID of zero assigns the Ethernet frame to the port/native VLAN.

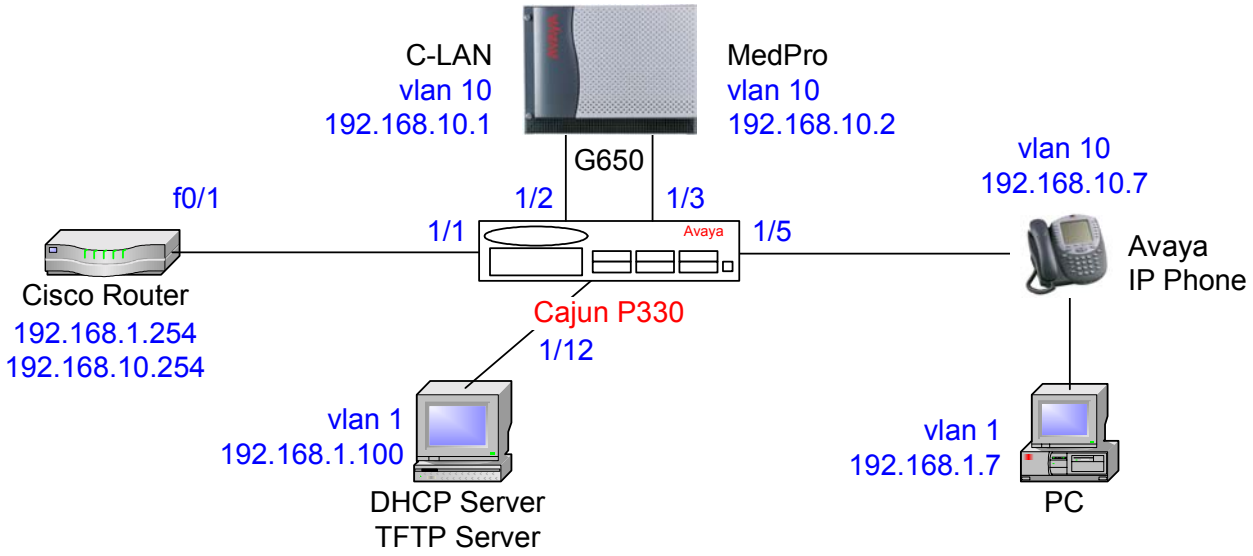
Cisco switches behave differently in this scenario, depending on the hardware platforms and OS versions. Here are Avaya Labs test results with a sample of hardware platforms and OS versions.

Catalyst 6509 w/ CatOS 6.1(2)	Accepted VID zero for the native VLAN when 802.1Q trunking was enabled on the port. In this case, all but the native VLAN should be cleared off the trunk.
Catalyst 4000 w/ CatOS 6.3(3)	Would not accept VID zero for the native VLAN. Opened a case with Cisco TAC, and TAC engineer said it was a hardware problem in the 4000. Bug ID is CSCdr06231. Workaround is to enable 802.1Q trunking and tag with native VID instead of zero. Again, clear all but the native VLAN off the trunk.
Catalyst 3500XL w/ IOS 12.0(5)WC2	Accepted VID zero for the native VLAN when 802.1Q trunking was disabled on the port.
Conclusion	Note the hardware platform and OS version and consult Cisco's documentation, or call TAC.

Note that setting a L2 priority is only useful if QoS is enabled on the Eth-switch. Otherwise, the priority-tagged frames are treated no differently than clear frames.

Sample Multi-VLAN Scenario for Avaya P330 Code 3.2.8 and Cisco CatOS and IOS

Here is a sample multi-VLAN scenario. Suppose there is a Cisco router connected to a P330 switch that contains two VLANs, one for the VoIP devices and one for the PCs. To conserve ports and cabling, the PCs are connected to the phones and the phones are connected to the P330 switch.



Cisco Router configuration	
interface FastEthernet0/1 **description 802.1Q trunk interface** **!** **interface FastEthernet0/1.1** **encapsulation dot1q 1** **ip address 192.168.1.254 255.255.255.0** **!** **interface FastEthernet0/1.10** **encapsulation dot1q 10** **ip address 192.168.10.254 255.255.255.0** **ip helper-address 192.168.1.100**	To forward DHCP requests to the DHCP server.

P330/P360 configuration (bind-to-static option)	
	All ports have port/native VLAN 1 by default.
set port vlan-binding-mode 1/1 static	Port in static binding mode by default, but command shown.
set port static-vlan 1/1 10	In addition to v1, v10 statically bound to port.
set trunk 1/1 dot1q	Port connected to Cisco router is an 802.1Q trunk port.
set port spantree disable 1/1	Spanning Tree disabled at the port level.
set port vlan 10 ½	Port/native VLAN changed to 10 on this port.
set port spantree disable ½	Spanning Tree disabled at the port level.
set port level ½ 6	Port L2 (802.1p) priority set to 6.
set port vlan 10 1/3	
set port spantree disable 1/3	
set port level 1/3 6	

set port vlan-binding-mode 1/5 static	Port in static binding mode by default, but command shown.
set port static-vlan 1/5 10	In addition to v1, v10 statically bound to port, but not a trunk port.
set port spantree disable 1/5	Spanning Tree disabled at the port level.
	Port 1/12 for the DHCP/TFTP server already has port/native VLAN 1.
set port spantree disable 1/12	Spanning Tree disabled at the port level.

P330/C360 configuration (bind-to-configured option)	
	All ports have port/native VLAN 1 by default.
set vlan 1	v1 configured
set vlan 10	v10 configured
set port vlan-binding-mode 1/1 bind-to-configured	Port bound to configured VLANs 1 and 10.
set trunk 1/1 dot1q	Port connected to Cisco router is an 802.1Q trunk port.
set port spantree disable 1/1	Spanning Tree disabled at the port level.
set port vlan 10 ½	Port/native VLAN changed to 10 on this port.
set port spantree disable ½	Spanning Tree disabled at the port level.
set port level ½ 6	Port L2 (802.1p) priority set to 6.
set port vlan 10 1/3	
set port spantree disable 1/3	
set port level 1/3 6	
set port vlan-binding-mode 1/5 bind-to-configured	Bound to configured VLANs but not a trunk port.
set port spantree disable 1/5	Spanning Tree disabled at the port level.

If the P330/C360 switch were a Cisco CatOS switch instead	
	All ports have port/native VLAN 1 by default.
set port host	First invoke this command on all user ports.
	Cisco switches do not tag the native VLAN, but the router expects a
set vlan 1005 1/1	tag on v1, so the native VLAN is changed to some unused VLAN.
set trunk 1/1 on dot1q	Port connected to Cisco router is an 802.1Q trunk port.
clear trunk 1/1 2-9,11-1004	Unnecessary VLANs removed; 1, 10, and 1005 remain.
set vlan 10 ½	Port/native VLAN changed to 10 on this port.
set port qos ½ cos 6	Port L2 (802.1p) priority set to 6.
set vlan 10 1/3	
set port qos 1/3 cos 6	
	Auxiliaryvlan is the more common method instead of explicit trunking.
set port auxiliaryvlan 1/5 10	v10 is the auxiliaryvlan; only v1 and v10 on this port; port is an 802.1Q trunk port, though not explicitly configured.
	Explicit trunking is an option.
set trunk 1/5 nonegotiate dot1q	Plain 802.1Q trunk port with no Cisco negotiation features.
clear trunk 1/5 2-9, 11-1005	Unnecessary VLANs removed; 1 and 10 remain.

If the P330/C360 switch were a Cisco IOS switch instead	
	All ports have port/native VLAN 1 by default.
interface FastEthernet0/1	

switchport trunk encapsulation dot1q	Port connected to Cisco router is an 802.1Q trunk port.
switchport trunk native vlan 1005	Cisco switches do not tag the native VLAN, but the router expects a tag on v1, so the native VLAN is changed to some unused VLAN.
switchport trunk allowed vlan 1,10,1005	VLANs 1, 10, and 1005 allowed on trunk.
switchport mode trunk	Port is in trunk mode.
spanning-tree portfast	Spanning Tree fast start feature.
interface FastEthernet0/2	
switchport access vlan 10	Port/native VLAN changed to 10 on this port.
spanning-tree portfast	Spanning Tree fast start feature.
switchport priority default 6	Port native VLAN L2 (802.1p) priority set to 6.
interface FastEthernet0/3	
switchport access vlan 10	
spanning-tree portfast	
switchport priority default 6	
interface FastEthernet0/5	
switchport trunk encapsulation dot1q	802.1Q trunk port.
switchport trunk native vlan 1	Since most PCs do not understand the tag, the PC's VLAN must be the native VLAN. v1 is already the native, but command shown.
switchport trunk allowed vlan 1,10	VLANs 1 and 10 allowed on trunk.
switchport mode trunk	Port is in trunk mode.
spanning-tree portfast	Spanning Tree fast start feature.
interface FastEthernet0/5	Simpler configuration on newer IOS switches (ie, 3550, 3560).
switchport mode access	Access mode; explicit trunking not required.
switchport access vlan 1	Configure the data VLAN.
switchport voice vlan 10	Configure the voice VLAN.

IP phone configuration

This procedure applies regardless of the Eth-switch used. Initially placing the IP phone on VLAN 10 requires two DHCP scopes – one for VLAN 1 and another for VLAN 10. Both scopes should have identical DHCP option 176 strings, with one exception. The VLAN 1 scope must have the **L2QVLAN** parameter, and the VLAN 10 scope should not. The following strings apply to phone firmware 1.8 and beyond.

VLAN 1:
MCIPADD=*addr1*,*addr2*, … ,HTTPSRVR=*addr*,L2QVLAN=10,L2QAUD=6,L2QSIG=6,VLANTEST=0
VLAN 10: **MCIPADD=*addr1*,*addr2*, … ,HTTPSRVR=*addr*,L2QAUD=6,L2QSIG=6,VLANTEST=0**

Run the phone through its normal boot-up sequence. It obtains an IP address on VLAN 1 – the port/native VLAN. When the phone receives the option 176 string above from the VLAN 1 scope, it releases the VLAN 1 address and enters a second DHCP sequence with tagging enabled to obtain a VLAN 10 address. After the phone is operational on VLAN 10, on subsequent reboots the phone returns to VLAN 10 directly, without passing through VLAN 1. In this example the **VLANTEST=0** option is invoked to make the phone permanently remain on the voice VLAN. See section 4.2, heading "IP Phone and Attached PC on Different VLANs" for a full explanation of how the phone operates between the data and voice VLANs, including the use of the **VLANTEST** parameter.

The **L2QVLAN** parameter should not be added to the VLAN 10 DHCP scope. This is so that in the event a phone is connected to a port that has VLAN 10 as the port/native VLAN, it will not receive instructions from the DHCP scope to enable tagging. In such a case the phone would not require tagging to function on VLAN 10, and tagging could result in an incompatibility with the Eth-switch.

PC configuration: The PC can be statically addressed with a VLAN 1 address, or it can receive a VLAN 1 address via DHCP. No special configurations are required.

Appendix B: Cisco Auto-Discovery

This appendix describes Cisco's proprietary auto-discovery feature using CDP and auxiliaryvlan or voice vlan, and how they relate to Avaya IP phones. Substantial testing and production operation have shown that Avaya IP phones interoperate with both **auxiliaryvlan** (CatOS) and **voice vlan** (IOS), and these have become the preferred methods of implementation over explicit 802.1Q trunking. This interoperability research was initiated because of the inability to enable **portfast** on older Catalyst 6500 code (pre 5.5.14, 6.3.2, 7.2.2) when the port is in trunk mode. The resulting request was to use auxiliaryvlan instead of explicit trunking, because portfast can be enabled on auxiliaryvlan ports, even on the older code releases.

Interoperability with auxiliaryvlan and voice vlan was successfully lab tested on the following platforms, with no known issues to date.

auxiliaryvlan on Catalyst 6509 w/ CatOS version 7.2.2

auxiliaryvlan on Catalyst 6509 w/ CatOS version 6.3.7

auxiliaryvlan on Catalyst 6509 w/ CatOS version 5.5.15

auxiliaryvlan on Catalyst 6509 w/ CatOS version 5.5.7a

auxiliaryvlan on Catalyst 6509 w/ CatOS version 5.5.3a

auxiliaryvlan on Catalyst 4000 w/ CatOS version 7.2.2

auxiliaryvlan on Catalyst 4000 w/ CatOS version 6.3.3

auxiliaryvlan on Catalyst 4000 w/ CatOS version 5.5.15

auxiliaryvlan on Catalyst 4000 w/ CatOS version 5.5.7a

voice vlan on Catalyst 3524 with IOS version 12.0(5)WC11

voice vlan on Catalyst 3550 with IOS version 12.1(22)EA1

voice vlan on Catalyst 3560 with IOS version 12.1x

Furthermore, Avaya IP phones have been deployed on a broader range of CatOS and IOS platforms by various Avaya customers, also with no known issues to date.

Therefore, auxiliaryvlan, voice vlan, and explicit 802.1Q trunking are all viable options when a dual-VLAN environment is required (see Appendix A). It is left to the user to choose the method, keeping in mind that auxiliaryvlan and voice vlan are Cisco proprietary mechanisms and are not subject to constraint by a standards body or by Avaya. 802.1Q trunking is well tested, successfully deployed, and defined by a standards body, but the configuration is not as clean, and trunking on user ports has other network implications.

For IOS-based Catalyst switches, voice vlan is roughly equivalent to auxiliaryvlan. On older IOS platforms (ie, 2900XL, 3500XL) there appears to be no configuration or functionality benefit to using voice vlan, as explicit trunking is still required when voice vlan is enabled on these older platforms. On newer IOS platforms (ie, 3550, 3560), however, voice vlan can be enabled without explicit 802.1Q trunking, so there are benefits to using voice vlan on these newer platforms.

Note that Avaya IP phones do not interoperate with CDP. Therefore, although auxiliaryvlan and voice vlan can be used, the mechanism of discovering these VLANs via CDP is not supported. The Avaya IP phone can learn the auxiliaryvlan/voice vlan designation via DHCP option 176, as explained below and in Appendix A.

How it Works

The remainder of this document focuses on auxiliaryvlan (CatOS), but voice vlan (IOS) operates on the same principles as auxiliaryvlan.

At the heart of Cisco's auto-discovery feature are Cisco-proprietary mechanisms. The first proprietary mechanism is CDP (Cisco Discovery Protocol). This is a layer 2 protocol, which means that it works at the Ethernet level, without requiring IP addresses. Cisco devices identify themselves to other Cisco devices using CDP packets that contain device- and port-specific information. (CDP packets can be captured and decoded using protocol analyzers that support CDP.) With the appropriate devices and OS versions, the CDP packets contain information specific to VoIP and other real-time applications. [1 p.2-22]

Using CDP, the Catalyst sends the Cisco IP phone an auxiliaryvlan ID, if auxiliaryvlan is enabled, and the phone tags its frames to be forwarded on that VLAN. The auxiliaryvlan is the second Cisco-proprietary mechanism, and it must be enabled on the port that connects to the IP phone. It is VLAN 200 by default or can be arbitrarily assigned as any number between 1 and 1000. According to Cisco's documentation the auxiliaryvlan is just another 802.1Q VLAN. The only difference is the proprietary method of assigning it to a Cisco IP phone. The port with the auxiliaryvlan also has a port/native VLAN (VLAN 1 by default or any arbitrarily assigned VLAN). This implies that the port is an 802.1Q trunk port with two VLANs, and can accept 802.1p/Q tagged frames. This is similar to the VLAN binding feature on the Avaya P330 v3.2.8 and later. [1 p.2-22, 2-23]

The information passed from the Cisco phone to the Catalyst is not of concern. The phone communicates its specific power requirements to the Catalyst, and the phone can also trigger the Catalyst to send its CDP packet immediately instead of waiting for the transmit period (60 seconds by default) to recycle. [1 p.2-23]

Avaya IP Phones on Cisco Auxiliaryvlan

The auxiliaryvlan is a modified method of implementing 802.1Q trunking, and it may be nothing more than this. Although testing to date has been positive, Avaya does not know what other mechanisms are or will be incorporated with this feature, or if they could have any adverse effects on Avaya IP phones. Assuming that an auxiliaryvlan-enabled port is truly a standard 802.1Q trunk port, the following steps allow Avaya IP phones to work on Cisco's auxiliaryvlan.

1) Verify that auxiliaryvlan is enabled.
   a) For example, the command **set port auxiliaryvlan 2/4-8 500** would make ports 2/4 through 2/8 auxiliaryvlan-capable with auxiliaryvlan ID 500.
   b) The command **set port auxiliaryvlan 2/4-8** (w/o the 500) would make ports 2/4 through 2/8 auxiliaryvlan-capable with the default auxiliaryvlan ID 200.
   c) The command **show port auxiliaryvlan** reveals the ports that have been made auxiliaryvlan-capable, and their respective auxiliaryvlan ID(s). The command **show port** reveals each port's port/native VID.
2) Bring up the phones on the auxiliaryvlan using the same procedures that would be used on a regular trunk port.
   a) Verify that a L3 router interface exists for both the port/native VLAN and the auxiliaryvlan, with an associated subnet and gateway IP address. Both interfaces must be configured to forward DHCP requests (**ip helper-address <*IP addr. of DHCP server*>**) to the DHCP server if the server is on a different subnet.
   b) Follow the instructions at the end of appendix A to get the IP phone on the auxiliaryvlan (voice VLAN).
   c) After the phone boots up, press **Hold ADDR #** to verify that the phone received an IP address and associated information for the auxiliaryvlan.
3) For call servers, IP boards (ie, C-LAN and MedPro/MR320), and other VoIP resources, configure their ports on the Eth-switch to be native to the auxiliaryvlan. That is, these ports do not require both

a port/native VLAN and an auxiliaryvlan. Just make the auxiliaryvlan the port/native VLAN on these ports (**set vlan 200 <mod/port>**, assuming 200 is the auxiliaryvlan ID). Then disable the auxiliaryvlan feature on these ports (**set port auxiliaryvlan <mod/port> none**).

4) Always verify network connectivity between devices using pings and trace-routes.

Appendix C: RTP Header Compression

RTP header compression is a mechanism that reduces the protocol overhead associated with VoIP audio packets. It is a function of the network and not a function of the VoIP application. Along with the benefits of using RTP header compression there are also cautions, and this appendix discusses both.

Application Perspective

Here is the anatomy of a 20-ms G.729 audio packet, which is recommended for use across limited bandwidth WAN links. Notice that two-thirds of the packet is consumed by overhead (IP, UDP, and RTP), and only one-third is used by the actual audio.

IP Header 20 B	UDP Hdr 8 B	RTP Header 12 B	20ms of G.729 Audio 20B

It is important to understand that all 20-ms G.729 audio packets, regardless of the vendor, are constructed like this. Not only is the structure of the packet the same, but the method of encoding and decoding the audio itself is also the same. This sameness is what allows an Avaya IP phone to communicate directly with a Cisco IP phone, or any other IP phone, when using matching codecs. The packets from the application perspective are identical.

Network Perspective

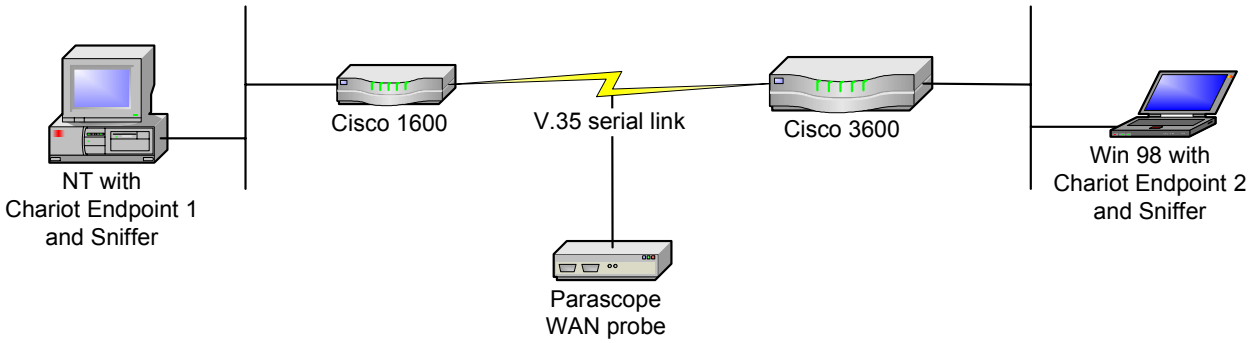
RTP header compression is a mechanism employed by routers to reduce the 40 bytes of protocol overhead to approximately 2 to 4 bytes [7 p.1] [2 p.5-14]. Cisco routers employ this mechanism, as does the Avaya X330WAN router, which is a module for the P330 chassis. RTP header compression can drastically reduce the VoIP bandwidth consumption on a WAN link when using 20-ms G.729 audio. When the combined 40-byte header is reduced to 4 bytes, the total IP packet size is reduced by 60% (from 60 bytes to 24 bytes). This equates to reducing the total VoIP WAN bandwidth consumption by roughly half, and it applies to all 20-ms G.729 audio packets, regardless of the vendor.

Customers who deploy routers capable of this feature may be able to benefit from it. However, Cisco recommends caution in using RTP header compression because it can significantly tax the processor if the compression is done in software. Depending on the processor load before compression, enabling RTP header compression could significantly slow down or crash the router. For best results, use a hardware/IOS/interface module combination that permits the compression to be done in hardware [3 QC-333] [5 "RTP Header Compression and QoS"].

RTP header compression has to function with exactness or it will disrupt audio. If for any reason the compression at one end of the WAN link and decompression at the other end do not function properly, the result could be intermittent loss of audio or one-way audio. This has been very difficult to quantify, but there is some anecdotal evidence. One production site in particular experienced intermittent one-way audio whose cause was very difficult to troubleshoot and isolate. When RTP header compression was disabled, simply for experimentation purposes, the audio problems went away.

The Test

This section details the results of a simple RTP header compression test conducted in a lab environment. Although this test was conducted using Cisco routers, the expected behavior is the same for any router that performs this function as specified in RFC 2508 [7]. This test was performed in the following lab configuration.



- NetIQ Chariot v4.0 was used to simulate VoIP calls between the two endpoints. Chariot v4.0 accurately simulates the characteristics of various codecs and uses a 40-byte IP/UDP/RTP header.
- Sniffer Pro v3.50.02 was used to capture the sent and received packets.
- The Cisco 3600 had IOS v12.1(2)T and the Cisco 1600 had IOS v12.0(12).
- The Fredericks Engineering Parascope WAN probe was tapped into the V.35 serial link to take bandwidth measurements.
- This test was performed using PPP encapsulation on the WAN link.

A single call was placed between the Chariot endpoints using the two most common codecs, sending 20-ms voice packets. Below are the results with and without RTP header compression. Note that these are rough measurements.

Codec	Payload bytes/packet	Packets/sec	Avg WAN BW consumption (kbps)		% reduction
			w/o compression	w/ compression	
G.711 (64 kbps)	160	50	84	68.5	~18 %
G.729A (8 kbps)	20	50	27.5	13	~53%

For each codec there was an attempt to verify that the audio packets were received in tact. This was done by spot-checking the audio packets before and after compression, using two Sniffer protocol analyzers. With G.729 the RTP header and payload were identical before and after compression. With G.711, however, the received packets had the PADDING flag set in the RTP header, although the flag was not set when the packets were transmitted. The PADDING flag indicates the presence of padding octets at the end of the RTP payload, which cannot be true for G.711. Why this occurred is unknown, but it does not really matter because there is no point in using the G.711 codec if bandwidth is scarce.

Configuration

To configure RTP header compression on a Cisco router,

1. Specify the number of RTP connections that can be compressed (cache allocation). In interface configuration mode, the command is **ip rtp compression-connections <number>**. The default is 32, and each call requires two connections. The configurable range is 3 to 256 for PPP and HDLC using IOS v11.3 and later; and 3 to 1000 for PPP and HDLC using IOS v12.0(7)T and later. For Frame Relay the value is fixed at 256.
2. The command to turn on compression is **ip rtp header-compression** in interface configuration mode. It must be implemented at both ends of the WAN link. For this experiment, when the command was entered into the router, **ip tcp header-compression** was also installed automatically. When either command was removed the other was automatically removed.

Consult Cisco's documentation for more specific configurations on other types of WAN links (ie, Frame Relay and ATM) [2 p.5-14, 5-18, 5-26, 5-33] [3]. Configuration for the X330WAN router is very similar to Cisco and well documented in the X330WAN User Guides.

Appendix D: Access List Guidelines

This appendix gives guidelines for configuring access lists to facilitate basic Avaya IP telephony functionality. The ports used by the Avaya call server are fairly fixed and known. The ports used by the endpoints are more variable and random. As a result, it is simpler to tailor access lists based on call server ports.

Action	From	TCP/UDP port or Protocol	To	TCP/UDP port or Protocol
The C-LAN uses UDP port 1719 for endpoint registration (RAS).				
Permit	Any C-LAN	UDP 1719	Any endpoint	UDP any
Permit	Any endpoint	UDP any	Any C-LAN	UDP 1719
The C-LAN uses TCP port 1720 for H.225 call signaling.				
Permit	Any C-LAN	TCP 1720	Any endpoint	TCP any
Permit	Any endpoint	TCP any	Any C-LAN	TCP 1720
This is to facilitate IP trunking between two Avaya call servers, and must be done for each IP trunk.				
Permit	Near-end C-LAN	TCP 1720	Far-end C-LAN	TCP 1720
Permit	Far-end C-LAN	TCP 1720	Near-end C-LAN	TCP 1720
This is one way to facilitate audio streams between MedPros/MR320s and endpoints.				
Permit	Any MedPro/MR320	UDP port range in ip-network-region form	Any endpoint	UDP any
Permit	Any endpoint	UDP any	Any MedPro/MR320	UDP port range in ip-network-region form
This is another way to facilitate audio streams between MedPros/MR320s and endpoints.				
Permit	Any MedPro/MR320	RTP/RTCP	Any endpoint	--
Permit	Any endpoint	RTP/RTCP	Any MedPro/MR320	--
This is to facilitate audio streams between direct IP-IP (shuffled) endpoints.				
Permit	Any endpoint	UDP any RTP/RTCP	Any endpoint	UDP any --
The R300 uses this default UDP port range for audio. However, the range is configurable.				
Permit	Any R300	UDP 1900-2075 RTP/RTCP	Any MedPro/MR320 or endpoint	UDP varies --
Permit	Any MedPro/MR320 or endpoint	UDP varies RTP/RTCP	Any R300	UDP 1900-2075 --
Permit	Any R300	UDP 1900-2075 RTP/RTCP	Any R300	UDP 1900-2075 --
These are all services used by the IP telephone. TFTP is tough to isolate to a port range. The GET and PUT requests from the client go to the server's UDP port 69, but all other messages go between random ports.				
Permit	Any IP telephone (hardphone)	UDP any	DNS server(s)	UDP 53 (dns)
Permit	DNS server(s)	UDP 53 (dns)	Any IP telephone (hardphone)	UDP any
Permit	Any IP telephone (hardphone)	UDP 68 (bootpc)	DHCP server(s)	UDP 67 (bootps)
Permit	DHCP server(s)	UDP 67 (bootps)	Any IP telephone (hardphone)	UDP 68 (bootpc)
Permit	Any IP telephone (hardphone)	TFTP	TFTP server(s)	--
Permit	TFTP server(s)	TFTP	Any IP telephone (hardphone)	--
Permit	Any IP telephone (hardphone)	TCP any	TLS server	TCP 411
Permit	TLS server	TCP 411	Any IP telephone (hardphone)	TCP any
Permit	Any IP telephone (hardphone)	TCP any	HTTP server	TCP 80-81
Permit	HTTP server	TCP 80-81	Any IP telephone (hardphone)	TCP any
Permit	SNMP management station(s)	UDP any	Any IP telephone (hardphone)	UDP 161 (snmp)
Permit	Any IP telephone (hardphone)	UDP 161 (snmp)	SNMP management station(s)	UDP any

Avaya devices ping other devices for various reasons. For example, C-LANs ping endpoints for management purposes; MedPros/MR320s ping C-LANs to gauge network performance across an IP trunk; IP telephones ping TFTP servers for verification purposes.				
Permit	Any Avaya device	ICMP Echo	Any	--
Permit	Any	ICMP Echo Reply	Any Avaya device	--

The following table contains access list guidelines for Avaya media servers and media gateways. Most connections take place over the S8xxx server's enterprise interface, which could be a separate interface or combined with a control network interface. The enterprise interface is…

- Eth4 on S87xx Multi-Connect.
- Typically eth0 on S87xx IP-Connect, but could also be configured as eth4 in some cases.
- Typically eth0 on S8500, but could also vary by configuration.

Action	**From**	**TCP/UDP port or Protocol**	**To**	**TCP/UDP port or Protocol**
This allows the Communication Manager 1.x primary server to synchronize translations with the S8300 LSP. A TCP session is initiated from the primary server to the LSP TCP port 514. A second session is then initiated from the LSP to the primary server TCP port range 512-1023.				
Permit	Primary server enterprise intfc	TCP any	LSP	TCP 514
Permit	LSP	TCP 514	Primary server enterprise intfc	TCP any
Permit	LSP	TCP any	Primary server enterprise intfc	TCP 512-1023
Permit	Primary server enterprise intfc	TCP 512-1023	LSP	TCP any
This allows the Communication Manager 2.x primary server to synchronize translations with the S8300 LSP.				
Permit	Primary server enterprise intfc	TCP any	LSP	TCP 21873
Permit	LSP	TCP 21873	Primary server enterprise intfc	TCP any
This allows the Communication Manager 3.x primary server to synchronize translations with the S8300 LSP.				
Permit	Primary server enterprise intfc	TCP any	LSP	TCP 21874
Permit	LSP	TCP 21874	Primary server enterprise intfc	TCP any
This allows an administrator to log in via Avaya SA to a call server (S87xx, S8500, S8300).				
Permit	Avaya SA workstation	TCP any	S8xxx enterprise interface	TCP 5023
Permit	S8xxx enterprise interface	TCP 5023	Avaya SA workstation	TCP any
This allows secure and unsecure web access to a call server (S87xx, S8500, S8300). The call server redirects unsecure sessions to https.				
Permit	Web admin station	TCP any	S8xxx enterprise interface	TCP 80
Permit	S8xxx enterprise interface	TCP 80	Web admin station(s)	TCP any
Permit	Web admin station	TCP any	S8xxx enterprise interface	TCP 443
Permit	S8xxx enterprise interface	TCP 443	Web admin station(s)	TCP any
Optional services used by call server (S87xx, S8500, S8300).				
Permit	S8xxx enterprise interface	UDP any	DNS server(s)	UDP 53 (dns)
Permit	DNS server(s)	UDP 53 (dns)	S8xxx enterprise interface	UDP any
Permit	S8xxx enterprise interface	UDP any	NTP server(s)	UDP 123 (ntp)
Permit	NTP server(s)	UDP 123 (ntp)	S8xxx enterprise interface	UDP any
H.248 signaling between G700/G350/G250 Media Gateway and S8300 or CLAN. MG initiates session.				
Permit	G700/G350/G250	TCP any	S8300 or CLAN	TCP 2945
Permit	S8300 or CLAN	TCP 2945	G700/G350/G250	TCP any
H.248 encrypted signaling between G700/G350/G250 Media Gateway and S8300 or CLAN. MG initiates session.				
Permit	G700/G350/G250	TCP any	S8300 or CLAN	TCP 1039
Permit	S8300 or CLAN	TCP 1039	G700/G350/G250	TCP any
Control network traffic and other traffic between S87xx/S8500 and IPSI board.				
Permit	S87xx/S8500 control interface	IP any	IPSI board	IP any
Permit	IPSI board	IP any	S87xx/S8500 control interface	IP any

Appendix E: Common IP Commands

Cisco CatOS Switches	
set port speed <mod/port> ?	sets the speed for given port(s)
set port duplex <mod/port> ?	sets the duplex for given port(s)
show port	displays settings and status for all ports
show port <mod/port>	displays settings, statistics, and errors for given port
clear counters ?	clears statistics and error counters on all ports or given port(s)
set port host ?	disables channeling/trunking; enables portfast on all or given port(s)
clear port host ?	opposite of **set port host**
set spantree portfast <mod/port> ?	enables or disables Spanning Tree fast start feature on given port(s)
show spantree [<mod/port>]	displays Spanning Tree and portfast info for all ports or given port(s)
set vlan <vlan id> <mod/port>	sets the native vlan (default vlan) for given port(s)
set port auxiliaryvlan <mod/port> <vid>	sets the auxiliary vlan for given port(s)
set port auxiliaryvlan <mod/port> none	removes auxiliary vlan from given port(s)
show port auxiliaryvlan ?	displays auxiliary vlan information
set trunk all off	disables trunking on all ports
set trunk <mod/port> ?	sets trunking mode for given port(s)
clear trunk <mod/port>	puts given port in auto trunk mode with negotiating encapsulation
clear trunk <mod/port> <vid(s)>	removes specified vlans from given trunk port(s) (all vlans are permitted on trunk by default)
show trunk ? OR show port trunk ?	displays trunking information for all ports or given port
show vlan ?	displays vlan configuration information

Cisco IOS Switches	
Global commands	
show running-config	displays all configurations currently running on switch
show startup-config	displays all configurations in NVRAM to be used at next boot-up
copy running-config startup-config	must be executed to save running configuration to NVRAM (not necessary on CatOS switches, except on router module)
show interfaces status	displays settings and status for all ports
show interfaces [fast|gig <mod/port>]	displays port(s) status, statistics, and errors at the interface level
clear counters [fast|gig <mod/port>]	clears **show interfaces** counters
show controllers ethernet-controller ?	displays port(s) statistics and errors at the controller level
clear controllers ethernet-controllers ?	clears **show controllers** counters
show vlan	displays vlan configuration information
Interface commands	These commands are executed on a port by port basis.
speed ?	sets the port speed
duplex ?	sets the port duplex
spanning-tree portfast	enables Spanning Tree fast start feature (no to undo)
switchport access vlan <vid>	sets the native vlan (default vlan) when port is in access mode (default is access mode, where there is only one vlan on port)
switchport mode trunk	puts port in trunk mode
switchport trunk encapsulation dot1q	makes trunk 802.1Q (instead of ISL)
switchport trunk allowed vlan ?	specifies vlans permitted on trunk port (default is all vlans)
switchport trunk native vlan	sets the native vlan (default vlan) when port is in trunk mode

Avaya P550/580 and P880/882 Switches	
show running-config	displays all configurations currently running on switch
show startup-config	displays all configurations in NVRAM to be used at next boot-up
copy running-config startup-config	must be executed to save running configuration to NVRAM (not necessary on P330 switches, except on router module)
set port auto-negotiation <mod/port> ?	enables or disables speed/duplex negotiation for given port(s)
set port speed <mod/port> ?	sets the speed for given port(s)
set port duplex <mod/port> ?	sets the duplex for given port(s)
show port status ?	displays settings and status for all ports or given port(s)
show port counters ?	displays high level TX and RX statistics for all ports or given port(s)
show ethernet counters ?	displays detailed statistics and errors for all ports or given port(s)
clear port counters ?	clears statistics and error counters on all ports or given port(s)
set port fast-start <mod/port> ?	enables or disables Spanning Tree fast start feature on given port(s)
show port [<mod/port>]	displays Spanning Tree and fast start info for all ports or given port(s)
set port vlan <mod/port> <vid>	sets the port vlan (default vlan) for given port(s)
set port trunking-format <mod/port> ?	sets trunking mode for given port(s)
set port vlan-binding-method <md/pt> ?	sets the vlan binding method for given port(s)
show port [<mod/port>]	displays trunking and vlan-binding info for all ports or given port(s)
show vlan ?	displays vlan configuration information

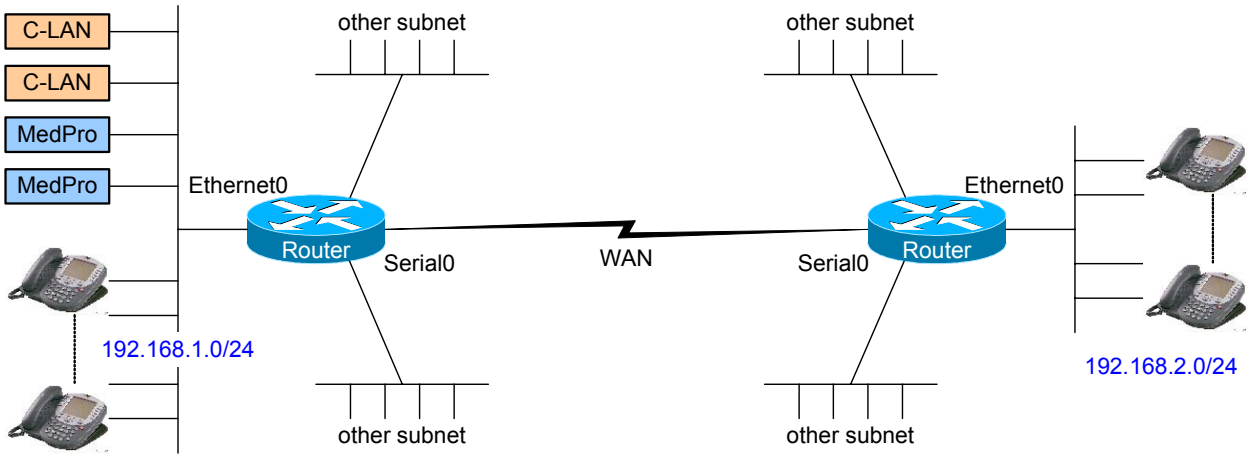
Avaya P330/C360 Switches	
set port negotiation <mod/port> ?	enables or disables speed/duplex negotiation for given port(s)
set port speed <mod/port> ?	sets the speed for given port(s)
set port duplex <mod/port> ?	sets the duplex for given port(s)
show port [<mod/port>]	displays settings and status for all ports or given ports(s)
show rmon statistics <mod/port>	displays statistics and errors for given port(s) (must reset switch to clear these counters)
set port spantree ? <mod/port>	enables or disables Spanning Tree on given port (no fast start on P330)
show spantree [<mod/port>]	displays Spanning Tree information for all ports or given port
set port vlan <vid> <mod/port>	sets the port vlan (default vlan) for given port(s)
set trunk <mod/port> ?	sets trunking mode for given port(s)
set port vlan-binding-mode <mod/prt> ?	sets the vlan binding mode for given port(s)
show trunk [<mod/port>]	displays trunking and vlan-binding info for all ports or given port(s)
show vlan ?	displays vlan configuration information

Avaya SAT and IPSI Interfaces	
change ip-interface <slot #>	sets the speed and duplex for an IP board
list ethernet-options	displays administered speed and duplex for all IP boards
get ethernet-options <slot #>	compares administered vs. actual speed and duplex for an IP board
IPSI commands	These commands are executed from the IPSI [IPADMIN] prompt.
set port negotiation 1 enable|disable	enables or disables IPSI control port (port 1) speed/duplex negotiation
set port speed 1 100MB|10MB	sets control port speed
set port duplex 1 full|half	sets control port duplex
show port 1	displays control port status and configuration
show control stats	displays control port statistics and errors

Appendix F: Sample QoS Configurations

This appendix gives simple examples of configuring QoS on Cisco routers. It is only meant to give the reader a starting point. Consult Cisco's documentation for a full explanation of Cisco's QoS implementation.

This rudimentary network configuration is used as a reference point. The objective is to assure high quality of service to VoIP applications across the congested WAN link.



Example 1 – Ideal / WAN terminating on G700/G350/G250 gateway Suppose all endpoints are capable of marking with one DSCP for audio and another DSCP for signaling. This would be true in an Avaya Communication Manager system with TN799DP C-LAN boards running firmware v5 or later. Previous firmware versions and the TN799C board cannot mark at L2 or L3. A matching set of configurations is applied to both routers.	
class-map match-any voipAudio **match ip dscp 46**	create a class map called voipAudio any packet with DSCP 46 is in this class
class-map match-any voipSig **match ip dscp 34** **class-map match-any ipsiSig** **match ip dscp 36**	create a class map called voipSig any packet with DSCP 34 is in this class create a class map called ipsiSig any packet with DSCP 36 (af42) is also in this class
policy-map voipQoS **class ipsiSig** **bandwidth 128** **class voipAudio** **priority 768** **class voipSig** **bandwidth 48**	create a policy map called voipQoS give packets in the ipsiSig class 128K of this WAN link reserve 768k of this WAN link for packets in the voipAudio class reserve 48k of this WAN link for packets in the voipSig class
class class-default **fair-queue** **random-detect dscp-based**	put everything else in the default class and transmit it out the default queue in a weighted fair queue fashion if the default queue starts to get full, randomly discard packets in this queue based on DSCP (lower values get discarded first)
interface Serial0 **description T1** **ip address 172.16.0.1** **service-policy output voipQoS**	apply the voipQoS policy outbound on this interface

This is an example of an ideal scenario where audio and H.323 and IPSI signaling are put in separate queues and receive different treatment. The X330WAN router and the G350/G250 integrated routers are optimized to use separate queues for audio and signaling. When terminating a WAN link to these devices, audio must be marked with DSCP 46, and signaling with DSCP 34 and IPSI signaling with 36.

Example 2 – Common large enterprise implementation

It is somewhat common to put audio and signaling in the same queue, in which case both audio and signaling would be marked with the same DSCP.

class-map match-any VoIP	create a class map called VoIP
match ip dscp 46	any packet with DSCP 46 is in this class
class-map match-any IPSI	create a class map called IPSI
match ip dscp 36	any packet with DSCP 36 (af42) is in this class
policy-map voipQoS	create a policy map called voipQoS
class IPSI	guarantee bandwidth to IPSI traffic
bandwidth 128	prioritize packets in the VoIP class and dedicate 816k of this WAN link
class VoIP	
priority 816	put everything else in the default class and transmit it out the default queue in a weighted fair queue fashion
class class-default	
fair-queue	if the default queue starts to get full, randomly discard packets in this queue based on DSCP (lower values get discarded first)
random-detect dscp-based	
interface Serial0	
description T1	
ip address 172.16.0.1	apply the voipQoS policy outbound on this interface
service-policy output voipQoS	

For applications where it is feasible, using a single queue for audio and signaling simplifies configuration and reduces router resource consumption (operating a single queue consumes less router resources than operating two queues).

Separate Queues vs. Single Queue

From a theoretical standpoint, using separate queues is ideal. When considering the three detriments to IP telephony – delay, jitter, and loss – audio is more sensitive to delay and jitter, whereas signaling is more sensitive to loss. This is not to say that audio is not sensitive to loss or that signaling is not sensitive to delay and jitter, but there are fine tuning points that apply to queuing to optimize it for audio or signaling. From a practical standpoint, in terms of user experience, these fine points may matter in some cases and not in others.

If the amount of signaling is negligible compared to audio, and if the size of the WAN link is such that serialization delay is not a factor (typically 768k or greater), then it is feasible to put audio and signaling in the same priority queue, as long as the queue is large enough to sustain both. In this case the larger signaling packets do not disrupt audio flow, because the serialization delay is low and because there are so few signaling packets relative to audio packets. Also, packet loss should not be an issue because the queue should be large enough to sustain both audio and signaling.

Suppose, however, that the ratio of signaling to audio is much greater – perhaps nearly 1:1. This would be possible in a remote office where all the signaling goes to a main office but most of the audio is local. Suppose also that the WAN link is relatively small (typically less than 768k) and serialization delay is a factor. In this case a large signaling packet entering the priority queue could delay audio packets, and even induce packet loss if the WAN link, and thus the priority queue, are small enough. It would be advisable in this case to use separate queues, optimized for the different characteristics of audio and signaling. As stated previously, the X330WAN router and G350/G250 integrated routers are optimized to use separate queues for audio (DSCP 46) and signaling (DSCP 34 or 41).

The preceding paragraphs are generalizations, and are not meant to imply a firm set of rules. Queuing is very complex, and implementations vary among manufacturers. The explanations given here are intended to give the reader a starting point. Testing with live traffic and real equipment, coupled with some trial and error, will ultimately dictate the optimum configuration for a given production environment.

Other Examples

Example 3	
Suppose that C-LANs 192.168.1.10 and .11 cannot mark their traffic (pre-Communication Manager system). This set of configurations is applied only to the left router.	
access-list 101 permit ip host 192.168.1.10 192.168.2.0 0.0.0.255 **access-list 101 permit ip host 192.168.1.11 192.168.2.0 0.0.0.255** Access list 101 permits any IP traffic from the two C-LANs to the 192.168.2.0/24 network. There is an implicit deny any at the end of this access list.	
class-map match-any untaggedVoIP **match access-group 101**	create a class map called untaggedVoIP packets matching access list 101 are in the class untaggedVoIP
policy-map setDSCP **class untaggedVoIP** **set ip dscp 46**	create a policy map called setDSCP for all packets in the class untaggedVoIP set the DSCP to 46
interface Ethernet 0/0 **service-policy input setDSCP**	apply the setDSCP policy inbound on this interface
Now the C-LAN traffic is marked with DSCP 46, as in example 2, and the example 2 configurations must be applied to both routers.	

Example 4	
This is the same as example 2, but with more restrictions on the traffic. In this example DSCP 46 is used throughout to simplify the access list. A somewhat matching set of configurations is applied to both routers.	
access-list 101 permit ip 192.168.1.0 0.0.0.255 192.168.2.0 0.0.0.255 dscp 46 (left router) **access-list 101 permit ip 192.168.2.0 0.0.0.255 192.168.1.0 0.0.0.255 dscp 46** (right router) Access list 101 permits any IP traffic that is marked with DSCP 46 between the two VoIP subnets. There is an implicit deny any at the end of this access list.	
class-map match-any VoIP **match access-group 101**	create a class map called VoIP only packets matching access list 101 are in the class VoIP; this is more restrictive than matching any packet with DSCP 46 the remainder of the configurations is identical to example 2
policy-map voipQoS **class VoIP** **priority 816** **class class-default** **fair-queue** **random-detect dscp-based**	create a policy map called voipQoS give strict priority to packets in the VoIP class on up to 816k of this WAN link put everything else in the default class and transmit it out the default queue in a weighted fair queue fashion if the default queue starts to get full, randomly discard packets in this queue based on DSCP (lower values get discarded first)
interface Serial0 **description T1** **ip address 172.16.0.1** **service-policy output voipQoS**	apply the voipQoS policy outbound on this interface
If any of the endpoints were incapable of DSCP marking, the "dscp 46" could be removed from access list 101. Then any traffic between the two VoIP subnets, regardless of the marking, would be in the class voip.	

Appendix G: IP Trunk Bypass – TDM Fallback Q&A

Q1: How does the IP trunk bypass (aka TDM fallback) feature work, and how should the parameters be set on the **system-parameters ip-options** form? How do these settings affect the IP trunk bypass feature?

The **system-parameters ip-options** form is used to define the thresholds that trigger a fallback to a TDM trunk, thus bypassing the IP trunk. For this feature to work, the 'Bypass if IP Threshold Exceeded' parameter must be set to 'y' in the **signaling-group** form for an IP trunk, and the correct route pattern must be administered. Simply stated, a near-end MedPro/MR320 monitors network performance by pinging the far-end C-LAN to measure network response against the configured thresholds.

One thing to note about the IP trunk bypass feature is that it is not fully supported on the S8300/media-gateway platform. The VoIP module in the G700 does not behave exactly like the MedPro/MR320 board, and it cannot perform the ping functions that a MedPro or MR320 performs. The issues with an S8300/media-gateway are discussed throughout this appendix.

When a high threshold is reached the signaling group goes into bypass state, and a fallback TDM trunk is utilized. When the corresponding low threshold is re-established the signaling group comes back into service, and the IP trunk is utilized. Because networks and user preferences vary, there is no single set of optimal thresholds. This is a feature that must be tested and fine-tuned with each implementation. The parameters are as follows.

- Roundtrip Propagation Delay (ms) High: 400-500ms is a good starting point for this threshold. Many users begin to notice performance degradation at around 200-250ms one-way delay.
- Roundtrip Propagation Delay (ms) Low: 200-300ms is a good starting point for this threshold. 100-150ms or less one-way delay typically results in very acceptable audio quality.
- Packet Loss (%) High: 7-10% is a good starting point for this threshold. Avaya Labs testing has shown that audio quality is acceptable even with 5% packet loss.
- Packet Loss (%) Low: 0-3% is a good starting point for this threshold.
- Ping Test Interval (sec): This is the frequency at which pings are sent out. The lower the interval the better for measuring network performance. In loads prior to Avaya Communication Manager 2.1 the low limit is 10sec, which is sufficient for detecting a network outage but not for measuring network performance. As of Communication Manager 2.1 and MedPro firmware v70 the minimum ping test interval is 1sec, which is granular enough to gauge network performance. 1-2 sec is a good starting point for this parameter.
- Number of Pings per Measurement Interval: This is the number of pings sent out before delay and loss are calculated. 10 should be used here for a minimum ping test interval of 10sec, which results in calculations every 100sec to detect a network outage. As of Communication Manager 2.1 and MedPro firmware v70, 20 to 30 pings at 1-second intervals results in calculations every 20 to 30 seconds, which provides the granularity required to gauge network performance.

Because pings are used to determine network performance, the IP network should ideally give the pings (ICMP Echoes and Echo Replies) between MedPros/MR320s and C-LANs the same priority as audio traffic. To facilitate this it is important to know that the call server can select any MedPro/MR320 in the near-end system's network region to originate the pings. Depending on the network, it may be feasible to activate this feature without deploying any network policies for the pings, especially if the primary concern is to compensate for network outages and not necessarily for poor performance.

Q2: Besides the IP trunk bypass feature, what other mechanisms are in place to detect an outage or severe congestion in the IP network, and how long does it take to detect it?

See section 3.5, heading "trunk-group and signaling-group" for details on the LRQ feature that applies to individual calls placed over an IP trunk. For the IP trunk as a whole the best method is the IP trunk bypass feature. In addition there is also a Maintenance Function. This function assesses the IP trunk every 15 minutes in a G3r or Linux platform, and every hour in a G3i platform. Without going into detail, the Maintenance Function determines whether the signaling group is in service or out of service. It can detect a network outage, but it does not assess network performance.

A third method was implemented as of Avaya Communication Manager 1.3. With this method a failure to set up a signaling link triggers the Maintenance Function to assess the IP trunk immediately. Assuming the failure to set up the signaling link is the result of a network outage, the Maintenance Function detects this and puts the signaling group out of service within one minute. For example, suppose there is an IP trunk between an S8700 system and an S8300/media-gateway. There is an outage in the IP network between the two systems and the S8700 discovers this after a measurement interval (IP trunk bypass feature). The S8700 puts the signaling group in bypass state and begins using the fallback TDM trunk. The S8300/media-gateway normally does not detect the outage until the next Maintenance Function cycle. However, if the S8300 attempts to place a call over the IP trunk and cannot establish a signaling link to the other end, this triggers the Maintenance Function immediately, which takes the signaling group out of service, causing the fallback TDM trunk to be used. So the S8300 detects the outage less than one minute after the first call attempt.

The scenario for severe congestion is different. In the case of severe congestion the S8700 detects the congestion and puts the signaling group in bypass state, the same as with a network outage. It then sends a message to the S8300 indicating this condition. (This message is also sent in the network outage case, but it doesn't reach the far end because of the outage.) The **status signaling-group** command at the S8700 shows the signaling group in bypass state. The same command at the S8300 shows the signaling group in far-end bypass state. In this condition both sides use the fallback TDM trunk until the S8700 puts the signaling group back into service.

Q3: As a follow-up to the previous question, what are the effects of the two sides not detecting the outage at exactly the same time?

Both sides accept incoming calls on TDM trunks, regardless of the state of IP trunks. So if side A detects an IP network outage and calls side B via the TDM trunk instead of the IP trunk, side B accepts the call. Side B continues to attempt using the IP trunk until it detects the outage, at which time it utilizes the TDM trunk for its outbound calls.

In the case of severe congestion, side A detects the congestion first, goes to bypass state, and starts using the TDM trunk. This causes side B to go to far-end bypass state and also use the TDM trunk. Eventually side B detects the congestion and goes to bypass state as well (unless the system is an S8300/media-gateway).

Q4: When the IP network recovers after an outage or severe congestion, do both sides discover this at the same time and start sending calls over the IP trunk at the same time? If not, what are the effects?

No, as with detecting the failure, detecting the recovery is also independent. But this is usually not a problem because both sides accept incoming calls on an IP trunk in bypass state. So if side A detects the IP network recovery first and calls side B while B is still in bypass state, side B accepts the call. However, the same is not true if side B is in out of service state.

The scenario for severe congestion is the same.

Q5: If the C-LAN or S8300 on one end of the IP trunk fails, does the IP trunk cover to a different C-LAN or S8300?

No, the IP trunk has fixed termination points. If one of the points fails the IP trunk goes out of service almost immediately at the local system where the failure occurred. This is especially true for an S8300 because it is the call server and not just a call signaling board like the C-LAN. At the remote system (the other end of the IP trunk) the IP trunk eventually goes out of service as follows. The IP trunk bypass feature puts the signaling group in bypass state (unless the system is an S8300/media-gateway). The Maintenance Function, either at the normal interval or triggered by a call attempt, puts the signaling group out of service. Depending on which of these occurs first the signaling group may go into bypass and then out of service, or out of service directly. A way to compensate for this type of outage is to administer multiple IP trunks (signaling groups and trunk groups) across multiple C-LANs between the same systems.

Q6: What about a MedPro/MR320 or VoIP module failure at either end of the IP trunk?

The IP trunk is not tied to any given MedPro/MR320 or VoIP module. As long as there is at least one MedPro/MR320 or VoIP module at each end with available DSP resources, the IP trunk is unaffected by MedPro/MR320 or VoIP module failures. If all usable Medpros or VoIP modules fail, the IP trunk's trunk group members go out of service, but the signaling group stays in service and can be used to send messages between the two systems. This essentially results in a bypass condition where the TDM trunk is utilized.

Q7: How is call processing affected in general by a C-LAN outage?

When configured properly the stations and media gateways have a list of alternate gatekeepers. They discover if a C-LAN they are registered with has gone down, and re-home to a different C-LAN. If the C-LAN failure occurs during an active call, the H.323 and H.248 link bounce recovery features preserve active calls on stations and media gateways, respectively.

Q8: How is call processing affected in general by a MedPro/MR320 or VoIP module outage?

The call server knows when a MedPro/MR320 or VoIP module has gone out of service and stops directing calls to that device. As long as there are sufficient MedPros/MR320s or VoIP modules to compensate for the outage, there is no adverse effect. If there is an outage during an active call, and that call is going through the affected MedPro/MR320 or VoIP module, that call loses audio. Avaya is studying the concept of redirecting an active call to a different MedPro/MR320 or VoIP module in this type of failure.

Q9: How is call processing affected in general by an IP trunk outage?

If the IP trunk outage is the result of a C-LAN/S8300 failure, direct IP-IP calls remain up until one of the IP phone goes on hook. If the IP trunk outage is the result of a MedPro/MR320/VoIP failure, existing calls are affected as previously described. If the IP trunk outage is the result of the IP network going down, the audio is lost on active calls, and new calls are routed over the fallback TDM trunk if one is administered.

Appendix H: IPSI Signaling Bandwidth Requirements

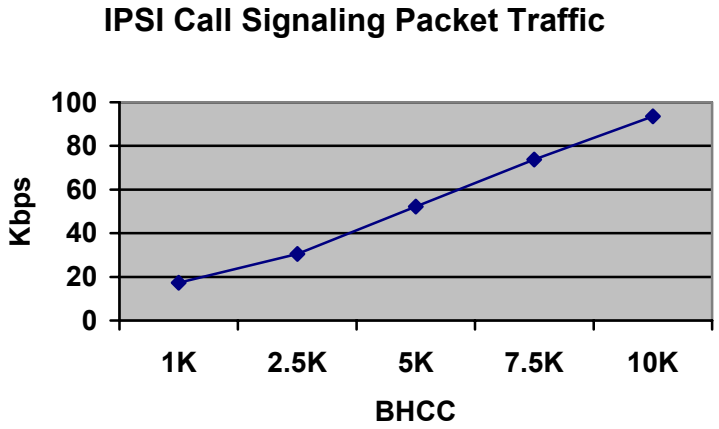
VoIP deployments require the provisioning of priority service for VoIP bearer traffic (RTP) and a guaranteed bandwidth service for Call Signaling. General VoIP requirements are discussed in the AVAYA IP Voice Quality Network Requirements document available on the www.support.avaya.com website. The following list summarizes key QoS requirements for voice traffic.

- Voice bearer traffic should be marked with DSCP 46 (EF).
- Voice bearer and Voice Signaling Packet loss should not be greater than 3%.
- One-way delay should be no more than 150 milliseconds.
- Average one-way jitter should be less than 30 milliseconds.
- H.323 Signaling Traffic should be marked DSCP 26 (AF31)
- IPSI Call signaling traffic should be given a guaranteed bandwidth on WAN links.
- IPSI Signaling Packet loss should not be greater than 3%.

The IPSI circuit pack provides enterprises with the capability to IP-connect Port Networks over LAN/WAN links in simplex and high availability configurations. Call signaling and system maintenance traffic is passed between S87XX/S8500 call control servers and the IPSI circuit packs in a port network. The call signaling traffic is encapsulated AVAYA proprietary CCMS (Control Channel Message Set) messages inside TCP/IP packets. The CCMS messages are H.323, H.225, and Q.931 messages used for registration of IP endpoints, to setup and teardown calls, periodic testing of the hardware, and keep-alive messages for IPSI connected port networks. These CCMS messages are critical to the stability of a port network and delivery of CCMS messages must be guaranteed.

The following table and graph displays IPSI call signaling traffic for varying Busy Hour Call Completion rates (BHCC). BHCC IPSI bandwidth is based on 150 IP endpoints originating and answering 10 second duration ISDN trunk calls within a port network. The simulated call scenario is a general business case. The common defaults for station traffic usage in a general business scenario are light traffic, moderate traffic**,** and heavy traffic.

BHCC Per PN	Usage Per Station	IPSI Bandwidth (Kbps) full duplex	IPSI TCP/IP packets per second
1K	Light Traffic	17.3	21
2.5K		30.5	37
5K	Moderate Traffic	52.2	61
7.5K		73.8	85
10K	Heavy Traffic	83.5	107



Provisioning for VoIP must include the Layer 2 overhead, which includes preambles, headers, flags, CRCs and ATM cell padding. The amount of overhead per VoIP call includes:

- Ethernet adds a 18 byte header, plus a 4 byte CRC plus an optional 4-byte 802.1Q Tag plus a 8 byte preamble for a total of up to 34 bytes per packet.
- Point-to-Point Protocol (PPP) adds 12 bytes of layer 2 overhead per packet.
- Multilink PPP adds 13 bytes per packet.
- Frame Relay adds 6 or 7 bytes per packet.
- ATM adds varying amounts of overhead depending on cell padding.
- IPSI encryption adds up-to 23 bytes (AES) for the encryption header and padding in addition to Layer 2 overhead.

IPSI bandwidth calculations should include the additional overhead on a per packet basis depending on the type of WAN link. For example; for a busy hour call completion rate of 5K calls (moderate general business traffic rate), the L2 overhead for a PPP link would be 61 PPS X 12 bytes/packet or 6.3 Kbps for additional PPP L2 overhead for a **minimum** of 58.1 Kbps. Encryption would add 23 additional bytes per packet or and additional 11.2 Kbps for a total of 69.3 Kbps.

A general rule of thumb for IPSI Control traffic bandwidth allocation is to add an additional 64Kbps of signaling bandwidth to the minimum required bandwidth in order to manage peak (burst) traffic loads and either round up or down to nearest DS0. Using the previous example of 5K busy hour calls using encrypted PPP links to control remote port networks you would guarantee 128Kbps (69.3Kbps + 64Kbps) for IPSI signaling bandwidth across the WAN link.

A standby IPSI consumes an additional 2.4 Kbps bandwidth on the standby link.

The following table summarized IPSI signaling bandwidth requirements.

BHCC	Ethernet	PPP	MLPPP	Frame Relay
1K	64Kbps	64Kbps	64Kbps	64Kbps
1K w/ encryption	64Kbps	64Kbps	64Kbps	64Kbps
2.5K	128Kbps	128Kbps	128Kbps	128Kbps
2.5K w/ encryption	128Kbps	128Kbps	128Kbps	128Kbps
5K	128Kbps	128Kbps	128Kbps	128Kbps
5K w/ encryption	128Kbps	128Kbps	128Kbps	128Kbps
>=7.5K	192Kbps	192Kbps	192Kbps	192Kbps
>=7.5Kw/ encryption	192Kbps	192Kbps	192Kbps	192Kbps

Cisco CBWFQ allows you to specify the exact amount of bandwidth to be allocated for a specific class of traffic. Taking into account available bandwidth on the interface, you can configure up to 64 classes and control distribution among them. Bandwidth can be assigned a percentage of total link speed or in Kbps. A FIFO queue is reserved for each class. Optional WRED can selectively discard lower priority traffic when the interface begins to get congested. Lower priority traffic should be guaranteed 25% of the available bandwidth on a WAN link.

Implement a QoS policies that provides:

- A queue for IPSI traffic using, for example, DSCP 36 (AF42) for IPSI signaling traffic. IPSI signaling can be assigned another DCSP value but must guarantee bandwidth to minimize Packet Loss.
- Expedited Forwarding - DSCP 46 - like behavior for the real-time voice. Call Admission Control (CAC) can be used to limit VoIP bandwidth.
- Assured Forwarding (AF31) like behavior for H.323 Call Signaling Traffic.

Note: DSCP 34 (AF41) is reserved for Video in the Cisco AutoQoS model but can be assigned for IPSI traffic when video is not deployed.

IPSI traffic classification can be assigned on a S87XX/S8500 via the **change ipsi-server interface** command. The **set diffserv 36** CLI command is used to mark traffic from IPSI to Server when you login to the IPSI. Use the **show qos** display the assigned marking.

It is important to note that H.248 Gateway Control traffic, as well as Call Center configurations, will have a greater signaling bandwidth requirement. Consult your account team for additional traffic requirements.

References

[1] Cisco Systems, Inc., "Cisco IP Telephony Network Design Guide," www.cisco.com, Customer Order Number: DOC-7811103=, Copyright 2001.

[2] Cisco Systems, Inc., "Cisco IP Telephony QoS Design Guide," www.cisco.com, Customer Order Number: DOC-7811549=, Copyright 2001.

[3] Cisco Systems, Inc., “Configuring Compressed Real-Time Protocol,” www.cisco.com, July 2002.

[4] Cisco Systems, Inc., “Troubleshooting Cisco Catalyst Switches to Network Interface Card (NIC) Compatibility Issues,” www.cisco.com, July 2002.

[5] Cisco Systems, Inc., “Understanding Compression (Including cRTP) and Quality of Service,” www.cisco.com, July 2002.

[6] IEEE, Inc., “802.1Q: IEEE Standard for Local and Metropolitan Area Networks: Virtual Bridged Local Area Networks,” www.iee.org, December 8, 1998.

[7] IETF, “RFC 2508: Compressing IP/UDP/RTP Headers for Low-Speed Serial Links,” www.ietf.org, February 1999.