Exam
Manual

LearnSmart

CISCO (642-436) CVOICE

Cisco Certified Voice Professional

Smarter Training

This LearnSmart exam manual covers the most important topics you will encounter on the Cisco Voice Over IP exam (CVOICE – 642-436). By studying this manual, you will become familiar with an array of exam-related topics, including:

- Components of a Gateway
- Dial Plan
- Basic Operation and Components Involved in a VoIP call
- And more!

Give yourself the competitive edge necessary to further your career as an IT professional and purchase this exam manual today!

CVOICE LearnSmart Exam Manual

Product ID: 11867
Production Date: July 19, 2011

Warning and Disclaimer

Every effort has been made to make this document as complete and as accurate as possible, but no warranty or fitness is implied. The publisher and authors assume no responsibility for errors or omissions. The information provided is on an "as is" basis. The authors and the publisher shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this document.

Volume, Corporate, and Educational Sales

Favorable discounts are offered on all products when ordered in quantity. For more information, please contact us directly:

1-800-418-6789
solutions@learnsmartsystems.com

International Contact Information

International: +1 (813) 769-0920

United Kingdom: (0) 20 8816 8036

Table of Contents

Abstract

The Cisco Certified Voice Professional is one of the most well respected certifications in the world. By attaining it, students and candidates signify themselves as extremely accomplished and capable Network Professionals. These four exams, created by Cisco Systems, are extremely difficult and not to be taken lightly. They cover a myriad of topics, in particular - the CVOICE (Cisco Voice Over Internet Protocol) exam covers all the way to the most detailed analysis of routing packets across wide area network in a complex, adaptable topology. CVOICE is multiple choice, simulative, and incorporates test strategies such as "drag and drop" and "hot area" questions to verify a candidate's knowledge.

Before taking this exam, you should be very familiar with both Cisco technology and networking. You must have also attained the Cisco CCNA certification by passing either the one or two part path.

Your Product

This CVOICE Exam Manual has been designed from the ground up with you, the student, in mind. It is lean, strong, and specifically targeted toward the candidate. Unlike many other CVOICE products, the LearnSmart CVOICE Exam Manual does not waste time with excessive explanations. Instead, it is packed full of valuable techniques, priceless information, and brief, but precisely worded, explanations. While we do not recommend using only this product to pass the exam, but rather a combination of LearnSmart Audio Training, Practice Exams, and Video Training, we have designed the product so that it and it alone can be used to pass the exam.

About the Author

Sean Wilkins is an accomplished networking consultant and has been in the field of IT since the mid 1990's working with companies like Cisco, Lucent, Verizon and AT&T. In addition to being a CCNP and CCDP, Sean is also a MCSE and an overall "IT expert". In addition to working as a consultant, Sean spends a lot of his time as a technical writer and editor.

Domain 1 - Describe the Components of a Gateway

Describe the Function of Gateways

The gateway concept is easy for almost everyone as it is simply a translator between one type of language or interface and another. The gateway is responsible for translating the information from the one to the other and vise versa. When talking specifically about an IP voice gateway the relationship is typically from an analog trunk/port or digital trunk/port to a packet based media. This media can be anything which can support IP from an Ethernet connection to a T1 which has IP running over HDLC/PPP. A simple example of this would be from a standard Foreign Exchange Station (FXS) analog port (what would connect to a typical analog telephone) to an Internet connection (Think Vonage). The IP gateway is responsible for taking the traffic from the analog FXS port and encoding it and transmitting it over the Internet connection running IP and in the reverse direction decoding the traffic from the Internet connection and formatting it over the FXS port.

The voice gateway also carries some other duties past simple translation; these involve support for various call control protocols, call setup and teardown, call hold, call transfer, DTMF relay among other duties. On Cisco equipment the call control protocols typically used are Media Gateway Control Protocol (MGCP), H.323, and Session Initiation Protocol (SIP) as well as various other protocols which are supplementary. Included with the support for these call control protocols is the responsibility of encoding and decoding a variety of different codecs which are used to efficiently send this traffic over a digital medium.

Describe DSP Functionality

As stated above one of the duties of the voice gateway is to translate the information from one type of media to the other. The Digital Signal Processors (DSP) job is to take an analog signal and translate it to a digital signal using a number of codecs into an IP stream. Part of this job is terminating the signal, sampling it and using the codecs to packetize it for transmission. This process also happens in reverse, as an IP voice stream comes in the traffic is terminated, decoded and translated onto the line. The gateway also has the capability to translate from one type of IP voice stream to another. The process of speaking between two different voice streams (Different call codecs, bandwidths, sampling rates…) is called transcoding and is also handled by the DSP inside the gateway. The DSP also has the ability to support voice conferencing which allows multiple callers to communicate over a common line.

Traffic Packetization

In order for an analog voice signal to be transmitted across a digital network, it must first be digitized. This is done through a process known as packetization. As shown in the following figure, packetization takes an analog waveform and converts it into a stream of digital 1's and 0's.

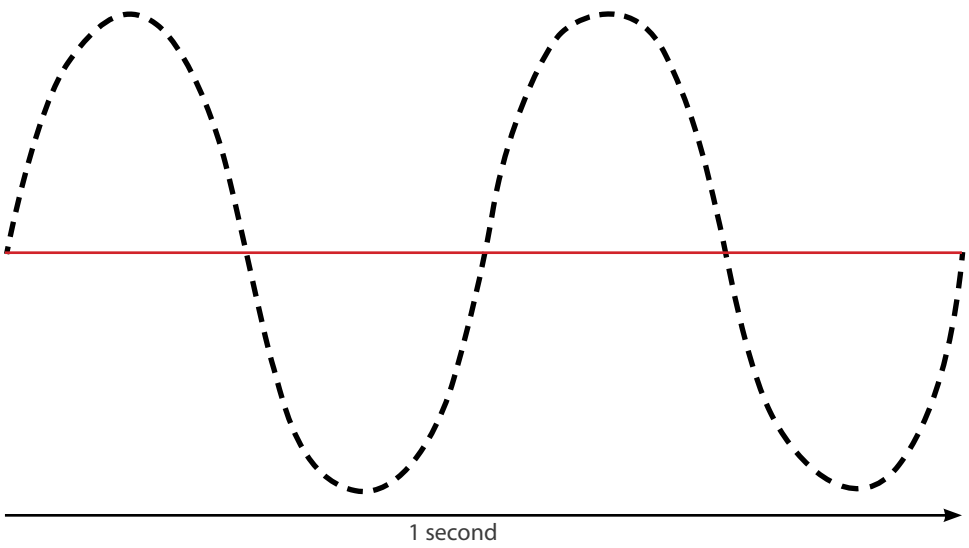


Figure 1 - Packetization

Sampling

The first step in digitizing a signal is to turn the analog wave into something that can be digitized. This is done through sampling. Sampling takes slices of the analog wave at consistent intervals. Within sampling, the Nyquist–Shannon sampling theorem is followed. This theorem states that in order to adequately digitally represent an analog signal, the analog signal must be sampled at a rate of twice the highest analog frequency. Within voice networks the frequency ranges from 300 to 3400 Hz is transmitted, because of simplicity it was decided to sample from 0 to 4000 Hz over digital lines. When following the Nyquist–Shannon sampling theorem, this means that this signal must be sampled at 8000 Hz which translates to 8000 samples per second.

In order to demonstrate this process the following figures have been created. Instead of showing 8000 Hz sampling the following examples show 40 Hz sampling, to keep it simple.

First an analog signal must be separated into pieces, because 40 Hz sampling is being shown, this analog signal is split into 40 different pieces (samples).

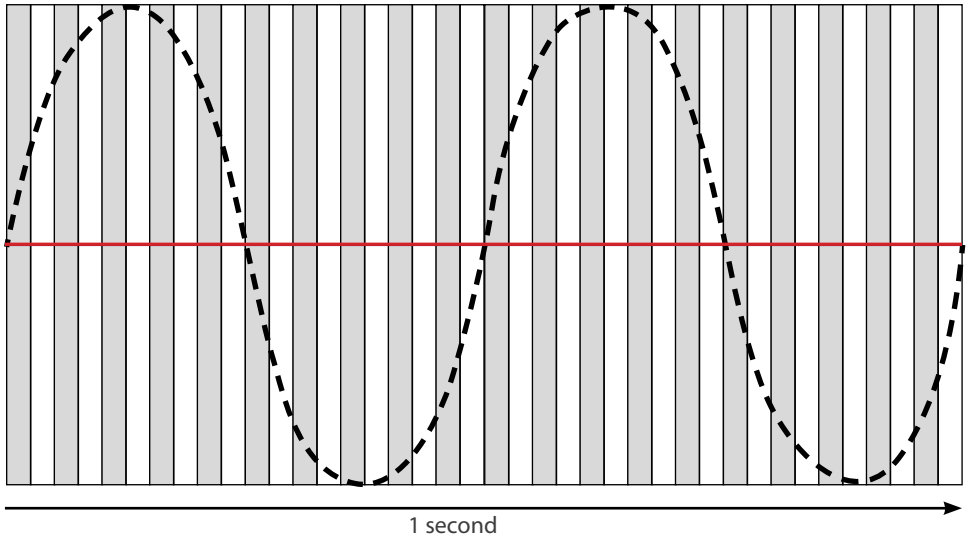


Figure 2 - Separating the Analog Signal

From these, pieces or samples are taken which best represent the analog signal; this is shown in the following figure:

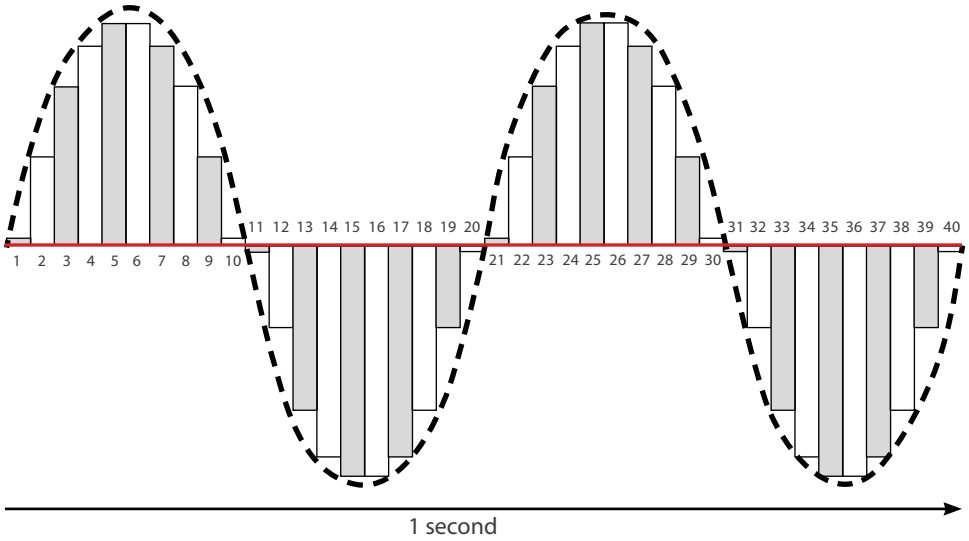


Figure 3 - Creating Samples

From this figure it is seen that the digital samples represent a signal similar to the analog signal being converted. Obviously, the more samples that are taken the more likely the original analog signal the digital representation will be.

Quantization

What this stage of processing does is it calculates a mathematical value for each sample taken, this is also called companding. For the purposes of this example we will describe Pulse Code Modulation (PCM) which is also referred to as G.711. With standard 64 kbps PCM the range of numbers that can be assigned is from -127 to +127 which are from the 8 bits used to record the signal, as seen from the following figure each sample is given a number. When the whole signal is given uniform translation regardless of the level of the signal it is called uniform (or linear) quantization. Uniform quantization results in low level signals having a higher Signal-to-Noise (SNR) ratio then higher level signals and because most signals are lower in nature this is rather inefficient. This problem is remedied with two different companding algorithms, μ-law and a-law.

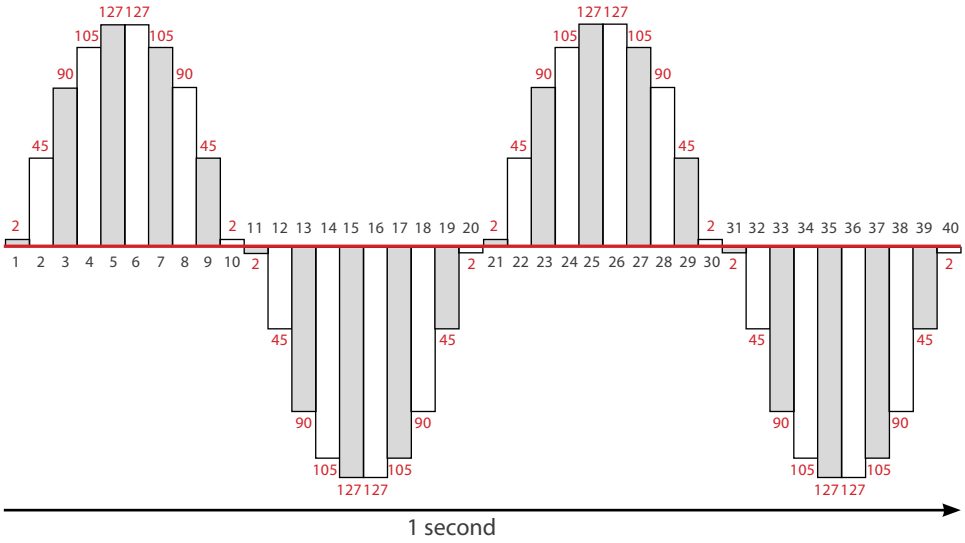


Figure 4 - Quantization

μ-law (mu-law) is the formal standard in North America and in Japan; a-law is what is used in the rest of the international community. These two algorithms work by taking a 14-bit (μ-law) or 13-bit (a-law) PCM sample and mapping it logarithmically to an 8 bit sample. Put simpler, this means that a larger signal is compressed down (14 or 13 bit) to fit in an 8-bit space and in order to remedy the linear quantization problem both μ-law and a-law encode lower level signals at smaller step intervals and higher level signals at higher step intervals. Both of these algorithms effectively increase the Signal to Noise (SNR) ratio of the signal. It is also standard in μ-law countries to convert to a-law in order to communicate with a-law countries.

In order to make these numbers into a stream which can be transmitted digitally as binary, encoding is needed. With PCM, the encoding process takes each number and converts it into a 7-bit binary number with the 1st bit being used to denote the sign (or *polarity*) with 1 meaning negative and 0 meaning positive, the 2nd, 3rd, and 4th bits signifying the *segment*, and the 5th, 6th, 7th and 8th bits signifying the *step*. Once the signal is converted to binary it is run through a digital to digital conversion process which shapes the signal for transmission. The figure on the next page shows this process:

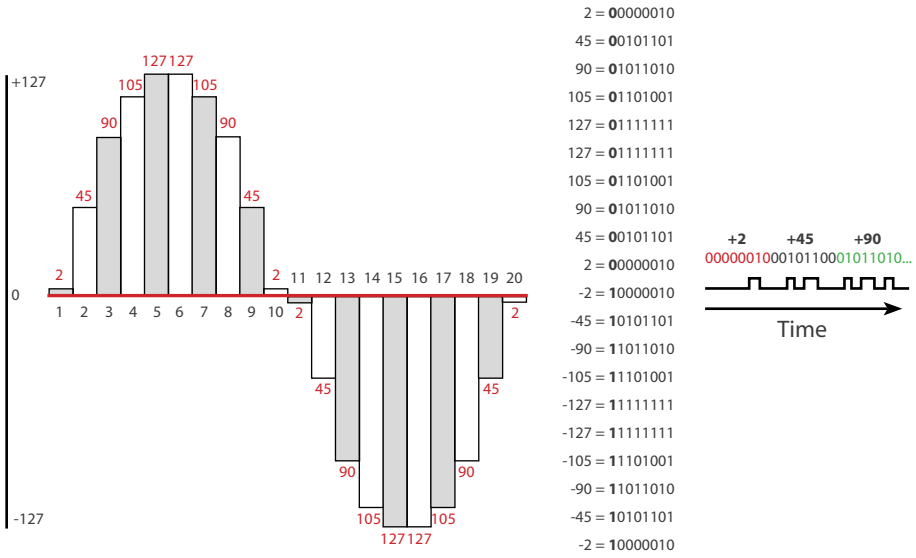


Figure 5 - Binary and Digital-to-Digital Encoding

Depending on the codec which is used on the signal this quantization phase operates in different ways in order to achieve bandwidth savings.

Describe the Different Types of Voice Ports and Their Usage

In order to communicate with any network, interfaces must be used. Within a VoIP network there are two different groups of interfaces; analog and digital.

Analog Interfaces

Inside voice networks there are three different types of analog interfaces; Foreign Exchange Office (FXO), Foreign Exchange Station (FXS) and Earth and Magneto (E&M). FXO and FXS interfaces are used with one another, the FXO interface is connected to the telephony switch and the FXS interface connected to the telephone equipment (phone). When a call comes in, the FXO interface is alerted via ring voltage from the switch then the FXO interface tries to transport the signal to the FXS. The FXS is responsible for receiving the signal from the FXO and providing battery, dial tone and other signaling to the telephone equipment. The E&M interfaces are typically used to connect Private Branch Exchanges (PBX) which exists inside offices. The PBX is essentially a small telephony switch that allows different features to be used inside an office environment; these types of features include extensions, forwarding, and conferencing among others. There are five different types of E&M interface; types I through V (1 through 5). The details of each interface are beyond the scope of this manual but types I and V are the most common; Type I is typical in North America and Type V is typical outside North America.

Analog Line and Trunk Signaling

There are five different types of line and trunk signaling: Loop-start, Ground-start, E&M wink-start, E&M immediate-start and E&M delay-start.

Loop-start signaling is the most common in a normal home telephone. With loop-start signaling the path from the telephony Central Office (CO) and the subscriber equipment is seen as a simple loop with voltage and ground provided by the CO. When the phone is on-hook the telephone will break this loop and thus no voltage will exist across the line. When the subscriber takes the telephone off-hook then the telephone will connect the loop and voltage will exists on the line, once this happens the CO switching equipment will see the voltage and offer a dial tone. If the phone is being called the CO will send a ringing voltage out the line, the telephone will see the differing voltage and initiate a ring. If the telephone is answered by coming off-hook then the telephone will connect the loop together, once this happens the CO switching equipment drops the ringing voltage and passes on the call. Loop-start works well for home services but is susceptible to *glare* which is what happens when both the switching equipment and the subscriber equipment try to come off-hook at the same time.

Ground-start signaling works a little differently. Before we describe this much more we must review some terminology; back in the days of phone operators there existed a type of wire connection which resembles a modern day ¼" audio jack (headphone). This type of connected had two points of connection, the *tip* and the *ring***.** In modern equipment, the *tip* typically is connected to the ground and the *ring* is typically connected to the voltage side. With ground-start there is a requirement for both sides of the connection to be separately grounded. In an idle or on-hook state the subscriber equipment has a break in the *ring* and the CO equipment has a break in the *tip*. When the subscriber side seizes the line it grounds the *ring* side of the connection, the CO equipment sees this and in response grounds the *tip* side of the connection. At this point the subscriber sees the grounded *tip* and connects the loop together and removes the ground from the *ring* side. When the CO equipment wants to seize the line it grounds the *tip* side of the connection, in response the subscriber equipment closes the loop and removes the *ring* ground connection. Ground-start signaling reduces the incidence of *glare* but does require a common ground.

E&M trunks are wired differently then FXS or FXO. The E&M line uses 8 wires and these wires are laid out as follows:

Lead	Description	Pin
SB	Signaling Battery	1
M	Signaling Input	2
R	Ring, Audio Input	3
R1	Ring, Audio Input/Output	4
T1	Tip, Audio Input/Output	5
T	Tip, Audio Input	6
E	Signaling, Output	7
SG	Signaling Ground	8

Table 1 - E&M Leads

E&M wink-start works by utilizing a pulse or *wink* of 140 to 290 ms to signal the receiving switch as ready for digits. When the originating switch seizes the line (typically using the M lead) it signals the remote side using the signaling leads as defined by the E&M type. The remote switch sees the line seizer on the E lead and transmits a *wink* by going off-hook from 140 to 290 ms, then returning to on-hook. The originating switch detects the *wink* then waits at least 100 ms and sends output digits.

E&M immediate-start works when the originating switch seizes the line and instead of waiting for a *wink* acknowledgement it waits a predetermined amount of time (> 150 ms) and transmits output digits. The remote switch only acknowledges the originating switch when the call is answered and the M lead is raised.

E&M delay-start works when the originating switch seizes the line by raising its M lead the remote switch acknowledges by raising its M lead. Once the remote switch is ready for output digits it will lower its M lead, once this happens the originating switch will send output digits.

Digital Interface

Within VoIP there are a couple of different digital interfaces; Basic Rate Interface (BRI), T1 and E1 being the main ones. A BRI is used for small office connectivity and provides two channels of voice (64 kbps each), which are called B-Channels and an independent signaling channel (16-kbps), known as the D-Channel. The data link layer of the D-Channel is Q.921 and the control signaling is typically done through Q.931. T1 signaling often gets confused because there are two different ways to signal with a T1 interface. The two different ways are Channel-Associated Signaling (CAS) and Common Channel Signaling (CCS). CAS utilizes what is called robbed-bit signaling, this is because T1's are divided into frames which fit into either a Superframe (SF – 12 Frames) or an Extended Superframe (ESF – 24 Frames). Each frame includes 24 timeslots which are used for the 24 T1 channels and each frame includes 8 bits of each channel plus a framing bit. In CAS signaling the 6th and 12th channels have their low order bit "robbed" for use in signaling. CCS signals a completely different way, it utilizes one full 64-kbps channel for signaling and leaves the other 23 channels for traffic; this configuration is typically called a Private Rate Interface (PRI). Having defined this, BRI is considered a CCS interface utilizing one full 16-kbps signaling channel. E1's all operate the same in a CCS configuration (sometimes they can be incorrectly called CAS). An E1 interface includes a total of 32 channels, with 30 being used for traffic. E1's are split similar to T1's except their framing is a little different. E1's are split into multiframes which include 16 contiguous frames and each frame includes 8-bits of each channel. The 1st and 17th channels are used for frame synchronization and signaling, accordingly. There are no framing bits within an E1 like T1's have.

An alternate to Q.931 exists called Q Signaling (QSIG) which is used as an interconnection between PBX, key systems and Cisco Call manager. QSIG consists of three sublayers: Basic Call, Generic Function and Supplementary services.

Describe Dial Peer Types

Call Legs

A short description of call legs must be understood before we move on to dial peers. A Call Leg is simply a logical connection between two telephony devices. Generally, this works by having an inbound and an outbound call leg central to the router. Dial Peers are the physical representation of the logical Call Legs, in a typical two-way connection there will be four call legs. When translating this to dial peers it would be a POTS Dial Peer on each router and two VoIP Dial Peers connecting the two routers together over an IP network.

Dial Peers

A Dial Peer is an addressable endpoint and is used to setup the parameters used to complete a call. These parameters can include calls based on called number, calling number, and entry port as well as a variety of different options which include codec used and IP destination. There can be multiple dial peers which can be matched based on many of these parameters.

There are four main dial peer types including POTS, VoIP, VoFR and VoATM although focus is typically put on POTS and VoIP. The POTS dial peers is what connects to a traditional PSTN network and is done through either analog or digital ports. The POTS peer is configured to provide an address(es) to be used for the port (telephone number(s)) and configures the physical port that is assigned these address(es). The VoIP dial peer is used to connect to IP Networks through any IP enabled interface or port. The VoIP dial peer is responsible for providing a routing mechanism for specifically configured addresses (telephone numbers); it also is used to configure the next hop router and specific parameters used on the VoIP connection.

Dial Peer Matching

Since there are several ways which a dial peer can be matched and since multiple dial peers can be configured concurrently it is important to understand the matching order which is used for dial peers. This matching is done a little differently depending on which direction a dial peer is being matched.

For inbound dial peers, the match is based on the source of the call. If a call comes in a voice port then the call is matched to a POTS dial peer and if a call enters in via IP then it will match a VoIP dial peer. If multiple Dial Peers exist which match the incoming call then an order is followed as to how to match a specific dial peer, these are as follows:

1. Match based on called number
2. Match based on calling number as configured with the answer address
3. Match based on calling number as configured with the destination pattern
4. Match based on the originating voice port
5. Match the "default" dial peer

The "default" dial peer should not be used in live networks because its parameters cannot be configured.

For outbound dial peers, the match is not based on the source of the call. Both POTS and VoIP dial peers are considered at the same time and will be match based on whichever one has the most specific called number match.

Describe Codecs and Codec Complexity

Compression is the main thing that has driven the use of VoIP. Through the use of compression, many voice calls can be conducted in the same amount of bandwidth as one had in the past. There are a couple of compression standards (International Telecommunication Union – ITU); these are called codec (Coder – Decoder) standards. The following table lists the standard codecs that are in use today:

Codec	Acronym	Name	Bit Rate
G.711	PCM	Pulse Code Modulation	64-kbps
G.722	SB-ADPCM	Sub-Band ADPCM	48, 56, 64-kbps
G.722.1	MLT	Modulated Lapped Transform	24 and 32-kbps
G.722.2	ACELP	Algebraic Code Excited Linear Prediction Coder	6.60, 8.85, 12.65, 14.25, 15.85, 18.25, 19.85, 23.05 or 23.85-kbps
G.723.1 (5.3-kbps)	ACELP	Algebraic Code Excited Linear Prediction Coder	5.3-kbps
G.723.1 (6.3-kbps)	MP-MLQ	Multi Pulse-Maximum Likelihood Quantization	6.3-kbps
G.726	ADPCM	Adaptive Differential Pulse Code Modulation	16, 24 and 32-kbps
G.728	LDCELP	Low Delay Code Exited Linear Prediction	16-kbps
G.729	CS-ACELP	Conjugate Structure Algebraic CELP (High Complexity)	8-kbps
G.729A	CS-ACELP Annex A	Conjugate Structure Algebraic CELP Annex A (Lower Complexity then G.729) (Medium Complexity)	8-kbps
G.729B	CS-ACELP	Conjugate Structure Algebraic CELP (G.729 with silence compression support) (High Complexity)	8-kbps
G.729AB	CS-ACELP	Conjugate Structure Algebraic CELP (G.729 A with silence compression support) (Medium Complexity)	8-kbps

Figure 6 - ITU Voice Codecs

Domain 2 - Describe a Dial Plan

Describe a Numbering Plan

At this point the easiest way to describe a numbering plan is to think about it in terms of an IP numbering plan. When building an IP network it is very important to layout the numbering of the whole network before deployment. If this action is not taken then the IP space will not be hieratical which makes routing and summarization on medium or large networks very cumbersome and troubleshooting almost impossible. A number plan in the voice world is just as important, think about the telephone number that you have right now. Now what if every person's phone number was a different area code and prefix, this addressing scheme, while possible, would be confusing for the end user. Now think about how this would be as the network operator, how would calls be correctly routed? For this to work every voice routing device would have to keep a complete routing table of every number, this is because simple summarization which happens in every Central Office (CO) would not be possible.

The International community follows a basic standard numbering scheme in order to interoperate, this is done through the E.164 standard. This standard lays out a one to three digit country code which is used by all countries to route calls to each individual country. Once this is done, each country is responsible for creating a national numbering plan; in North America this is done through North American Numbering Plan (NANP). To all who are familiar with North American number this will be familiar. The NANP follows a 10 digit numbering plan with a three digit Numbering Plan Area (NPA) which is used to identify an area, which is also why it is called the area code. A three digit code which is then used to identify a central office or exchange, this is also referred to as the NXX. And the final piece is the four digit station code which defines the specific phone or location.

Describe Digit Manipulation

In general terms, the purpose of digit manipulation is to add, change or delete the digits of a telephone number. There are a couple of different techniques which are used which include Digit Stripping, Forward Digits, Prefix Digits, Number Expansion, Calling Line Identification (CLID) and Voice translation profiles. The following sections describe these different techniques.

Digit Stripping

Digit Stripping is a technique which is specific to POTS dial peers; this is because by default POTS dial peers strip any digits which are matched to the destination pattern. This is different from VoIP dial peers because VoIP dial peers do not strip any digits by default. Now for the POTS peers this means that if a specific number pattern needs to be matched but also some or the entire matched digit needs to be translated outbound then Digit Stripping must be disabled. Digit Stripping can be helpful in some circumstances because a specific matched pattern to always be stripped is wanted. A good example of this is phones which are required to dial a specific digit to gain access to an outside line, if 9,212-555-1212 is dialed, the 212-555-1212 needs to be translated to the outside switch but the 9 is not to be translated. Digit Stripping on the POTS peers would do this by default if the match was based on the 9 starting an external call. If Digit Stripping was to be turned off the same configuration would translate the whole number (9,212-555-1212) out to the external switch.

Forward Digits

The configuration of the forward digits tells a POTS dial peer how many called party digits to send out to the peer. If the number of digits which are input is larger then the number of digits configured to be forwarded then the rightmost digits are sent. For example, if the number 212-555-1212 was sent and the equipment was configured to forward seven digits then only the 555-1212 would be sent. The Forward Digits can also be configured to send extra digits if input, what this does is allow digits longer then the destination pattern to be sent as well.

Prefix Digits

The purpose of the configuration of prefix digits is to add additional digits to a called number. Utilizing prefix digits like forward digits is limited to POTS dial peers. When prefix digits are added it's very important, once a call has been entered and matched an outbound dial peer AND after Digit Striping has occurred is when Prefix Digits are added to the called number.

Number Expansion

Number Expansion is very similar to Prefix Digits as digits are configured to be added to the called number. The big difference between Prefix Digits and Number Expansion is that Prefix Digits are configured on the POTS dial peer and Number Expansion is configured on the whole gateway. The digit addition happens before the outbound dial peer. Number Expansion changes how the outbound dial peer is configured because the dial peer must match the dialed number after the Number Expansion has happened.

CLID

Calling Line Identification or CLID is used to display the calling telephone number on the called party's equipment. This CLID can be manipulated in a number of ways. The CLID can be stripped completely, left intact but do not display, and change the number completely among others.

Voice Translation Profiles

Voice Translation Profiles are used in a much more flexible way. These profiles are allowed to be associated with a dial peer, voice port, trunk group or globally on all VoIP calls. This flexibility is configured through translation rules. Multiple translation rules can be configured for each translation profile.

Depending on where the Voice Translation Profile is configured effects how the specific rules are applied. If a Voice Translation Profile is applied at the voice port then the rules are applied before the inbound dial peer is matched. If a Voice Translation Profile is applied to an inbound dial peer then the rules are applied before the outbound dial peer is matched. If a Voice Translation Profile is applied in an outbound dial peer then the rules are carried out before a call is transmitted.

Describe Path Selection

Hunt Groups

A Hunt group has two main purposes, to load balance among dial peers or to backup a dial peer or number of dial peers. A Hunt Group is composed of multiple dial peers who have the same destination pattern. The destination pattern does not however have to be exactly the same but is meant to be for the same destination. Dial peer hunting is enabled by default.

With a Hunt Group there is a selection order which is followed by the gateway in order to find the best dial peer match. This order is as follows; longest match, preference value, and random selection.

The longest match is simple; the dial peer with the most specific destination pattern will be preferred over all others. For example, if a destination pattern was given of 555…. then the dial peer would match based on a number which included 555 and four other numbers. If a destination pattern was given of 5551… then the peer would match based on a number including 5551 and three other numbers. Both of these destination patterns could be in the same hunt group but any number which matched the 5551… pattern would prefer the most specific destination pattern match.

Hunt groups also have a preference value which can be given to specific members of the group to assign priority. This priority value can be from 0 through 10 with the lower number being preferred; by default 0 is given to all dial peers. If the dial peers being matched have the same destination pattern then the dial peer with the lowest preference value will be preferred.

If there were two dial peers who have the exact same destination pattern then equal preference is given to each dial peer and the different members of the group would load balance between each other in random order.

If the hunt process needs to be stopped, this capability exists on each dial peer.

Trunk Groups

A Trunk Group works on voice ports including BRI, PRI, CAS, FXO, FXS and E&M. A Trunk Group is essentially a hunt group of parallel trunks or ports. A number of ports can be put into a trunk group, a variety of parameters can be assigned dictating the number of calls allowed both incoming and outgoing, and trunk selection among others. Trunk groups are used through the dial peer, the dial peer points to a trunk group and the trunk groups points to, configures and chooses the port to take the call.

CAC

Call Admission Control (CAC) also has the capability to control how a call path is selected. The purpose of CAC is to limit the number of calls which are able to go over an IP connection. This is because VoIP paths do not have the same physical call restrictions which exist on other ports. With an IP connection a call will be routed without CAC for as many calls as are possible through the existing ports, even if this seriously degrades the quality of the calls already established. CAC prevents this through the use of a mechanism for controlling the number of calls based on the available bandwidth. These mechanisms can use Quality of Service metrics to come up with an allowed amount of calls which can be established at one time. There are a number of different QoS mechanisms, including Integrated Services (Intserv) and Differentiated Services (Diffserv).

Describe Calling Privileges

The concept of calling privileges is simple; what places are the phones on your network allowed to place and receive calls from. Calling privileges through the use of Class of Restrictions (COR) is able to make this distinction of which of these calls is possible. COR creates a simple lock and key mechanism which is put in place on the dial peers. The "Lock" is typically put on an outgoing dial peer while the "Key" is typically put on the incoming dial peer. If an incoming call does not come from a dial peer which has the right "Key" then the call will be blocked.

There is an important caveat to COR, that is that if there is no incoming COR list associated with an incoming dial peer then the call will succeed regardless. Simply stated, an incoming COR list MUST be assigned to an incoming dial peer for the lock and key mechanism to work as planned.

Describe Call Coverage

Call coverage is a number of mechanisms which can be used to ensure that a call is answered. These methods include Call forward, Call hunt, Call pickup, Call Waiting, and Hunt Groups among others. Call forwarding is a rather simple concept which involves a number being called to forward to another based on a busy signal, on no answer, and at night among others. Call hunt works by passing a call from one extension to another until someone finally answers the phone. Call pickup gives the ability for other people who are not being directly called to pick up a line which is being called. Call waiting gives the ability to show the called party an incoming call and to let the party decide how to deal with the call and Hunt Groups are described in more detail above.

Domain 3 - Describe the Basic Operation and Components Involved in a VoIP Call

Describe VoIP Call Flow

In general terms, the flow of a call includes everything from the initiation of the call through the release of disconnection of the call when complete. Each of the three main VoIP protocols handles their call flows in different ways; because of this more specifics of each protocols call flow will be detailed in the following specific sections. The following sections describe the common stages of the VoIP call flow.

Call Stages

Call Setup

As the name implies this is the stage that sets up the call. This stage can be broken down into the initial call request, call proceeding, call progress, alerting and the connection. The initial call request involves looking up the telephone to IP conversion for call routing, and trying to send a request to the called party. Now a call can fail at this point for a number of reasons, including no response to the lookup (don't know where to go for the number), no resources available (no bandwidth or trunk space for the call), or no response from any of the nodes inline to the called party. Call proceeding notifies the calling party that the call is going through. Call progress notifies the calling party of the status of the call. Call alerting tells the calling party the status through the use of sounds (busy, fast busy…) and the connection (or Connect) is when the call has been answered, parameters are negotiated and the connection is established. The main negotiated parameters include ports used for the call and the codec.

Call Teardown

Call teardown is easy, a request is sent from either side of the connection to terminate. The switch notifies all the switches inline from the calling to the called party to teardown the connection. Once this is complete there is no connection left.

Call Maintenance

Call maintenance can be done by any part of the voice network. The switch can collect data on all calls and connections to and through it. The phones and other endpoints can collect information about the call including call statistics, quality, delay and jitter among other things.

Describe RTP, RTCP, cRTP, and sRTP

Real-Time Transport Protocol (RTP)

All voice over traffic (not signaling) is carried via UDP; however UDP by itself has some problems that need to be addressed for Voice over IP service to work. UDP by itself does not have packet sequencing and reordering capabilities or the ability to time stamp. Because of these shortfalls another protocol, Real-Time Transport Protocol (RTP) was created to run over UDP and provide these capabilities. These capabilities on top of UDP's ability to multiplex traffic make voice over technologies work well and keep a high quality level. RTP is used with all of the VoIP protocols. The problem with RTP is that it's header adds extra overhead to the voice packet, 12 bytes in total for voice with an additional 60 bytes possible if optional headers are used for other types of media streams.

RTP Control Protocol (RTCP)

RTP also has a monitoring protocol that works with it called Real-Time Transport Control Protocol – (RTCP). RTCP is used to monitor the RTP session and update the participants of the status of the stream. This functionality is typically used for Quality of Service (QoS). The devices which control the QoS on the gateways use this information to control the flow of RTP. The traffic service parameters could need to be changed or the codec might need to be changed based on available RTCP data.

RTP Header Compression (cRTP)

While RTP provides extra needed services that UDP does not, it also provides added overhead. So with RTP the total overhead of an IP packet is 40 bytes, 20 bytes for the IP header, 8 bytes for the UDP header and 12 bytes for the RTP header. What cRTP provides is a way to minimize the header overhead. cRTP does this by assigning a hash to the IP, UDP and RTP headers, which then replaces the IP, UDP and RTP headers completely. Since these headers for the duration of a call are exactly the same, the hash is used to replace them until the headers change, if the headers do change cRTP sends the full headers again. This hash without checksum is only 2 bytes saving 38 bytes of overhead per packet, with checksum the hash is 4 bytes saving 36 bytes of overhead per packet. cRTP is applied per link and is required to be run on both sides of the link for it to work correctly. It is the recommendation of Cisco that this only be used on connections below 2 Mbps when cRTP is performed in hardware and below 768-kbps if in software. cRTP does add delay when computing the hashes and should be considered when calculating the overall expected delay. By default, Cisco places a limit of 16 concurrent cRTP sessions per link; this can be changed if the resources are available.

sRTP

Secure RTP is in implantation of RTP which allows the payload of RTP to be encrypted between devices. This can work from end to end if all equipment supports it and does require that all components along the path support sRTP. sRTP works by encrypting the payload of the RTP packet with Advanced Encryption Standard (AES) encryption. The keys for the encryption are passed in the clear before the transmission starts and only the payload is encrypted the headers are not. If there is a need for the keys or the header to be encrypted as well then an implementation of IP Security (IPSec) is needed. At this time sRTP is supported on MGCP, SIP and H.323 equipment but in some cases not fully, please read up on the current Cisco IOS release notes for updates.

Describe H.323

Components

H.323 Gateway
An H.323 gateway can be used to translate between different types of connections. With H.323 this includes analog phones, trunks, and various other interfaces. The H.323 gateway also has the capability to place and receive calls without the use of a gatekeeper or with a gatekeeper.

H.323 Gatekeeper
The H.323 gatekeeper acts as a central point for resolving H.323 phone numbers and IP addresses and also to control admission control. The gatekeeper also takes care of call routing and control, security and bandwidth management.

H.323 Terminal
The H.323 terminal is a device which can do any real-time two-way communication. These devices include IP phones, conferencing equipment, gateways and Call Managers.

H.323 Call Flow

H.323 instead of defining a specific protocol specifies a group of protocols which are used to connect devices together through a distributed model. With H.323 this includes not only voice services but also video. With H.323 there are four phases to creating a connection, this includes an admission request, connection setup, capabilities exchange and the opening of the media.

In the admission request phase the endpoint or gateway communicates with the gatekeeper via H.225 over UDP. The gatekeeper checks to see if it knows how to route the call requested, if the call route exists the gatekeeper checks if a path is available from the 1st to the 2nd endpoints with available resources. If the resources are available then the gatekeeper performs an address translation from phone number to IP address and returns the information to the requesting endpoint. The admission is done through Registration, Admission and Status (RAS) messages. For admission this is done through Admission Requests (ARQ) and Admission Confirm (ARF) messages.

During the connection setup, the endpoints communicate directly via H.225 over TCP. During this the endpoints take the lookup information and establish a connection to the endpoint. During this connection a setup message is sent from the originating endpoint, if the call is accepted a connect message is exchanged.

During the capabilities exchange the endpoints communicate directly via H.245 over TCP. Capabilities which are exchanged include voice or video communications, codec exchange, compression exchange and coding exchange including others. It is at the end of this phase that the end user is finally notified of an incoming call.

Now the final phase includes the opening of the media via H.225 over TCP. If the end user has answered the call connect messages are used to open the connection.

Once the call is complete then the call is released using H.225, this is done through Disengage Requests (DRQ) and Disengage Confirm (DRF) messages.

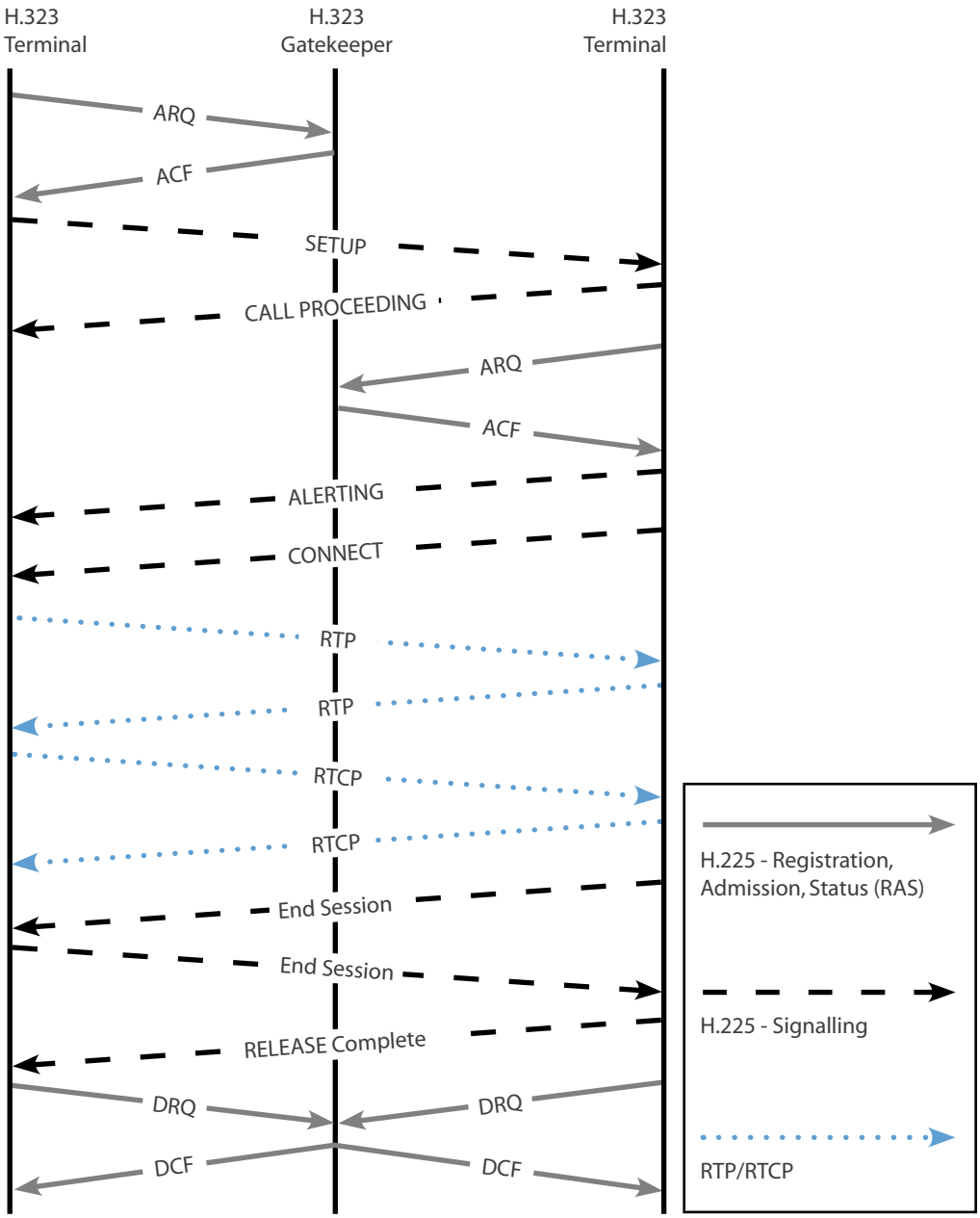


Figure 7 - H.323 Call Flow

Describe MGCP

The Media Gateway Control Protocol (MGCP) works a little different then the other two protocols. MGCP works by controlling multiple gateways. Within MGCP there are two main components, the Media Gateway (MG) and the Media Gateway Controller (MGC). The MGs are responsible for connecting and translating connections into the network. These can exist anywhere from the end user's house to a central telephone office. The MGC is the call agent and is responsible for call control; for more information on Call Agents see the Central call control model section. With MGCP if the connection to the MGC is removed then the gateways do not know how to independently route calls; this functionality is completely up to the MGC.

The MGCP there are two main types of gateway, a residential gateway and a trunking gateway. Residential gateways handle the interfaces like analog ports and VoIP networking interface. The trunking gateways handle the interfaces which go out to the PSTN network.

MGCP Messages

There are nine different MGCP messages which are used to communicate between the gatekeepers, gateways and endpoints. These messages are:

Message	Message Name	Used by	Description
AUEP	AuditEndpoint	MGC	Determines the status of the given endpoint
AUCX	AudioConnection	MGC	Retrieves the parameters associated with the connection
CRCX	CreateConnection	MGC	Creates connection between endpoints
DLCX	DeleteConnection	MGC & MG	MGC: Terminates connection MG: Indicates connection can't be sustained
MDCX	ModifyConnection	MGC	Changes the parameters of a connection
RQNT	NotificationRequest	MGC	Instructs the gateway to watch for special events (off-hook, DTMF, etc). Also used to communicate to the gateway to provide a signal (Dial Tone, Busy, etc)
NTFY	Notify	MG	Informs MGC when events occur
RSIP	RestartInProgress	MG	Informs MGC that a group of endpoints has been taken out of service or put back into service
EPCF	EndpointConfiguration	MGC	Instructs the gateway to provide specific configuration to the endpoint

Figure 8 - MGCP Messages

MGCP Call Flow

We must establish that the gatekeepers in the case of Cisco are Cisco Call Manager and the gateways can be various network equipment which supports MGCP voice. The following shows a call from one analog phone to another:

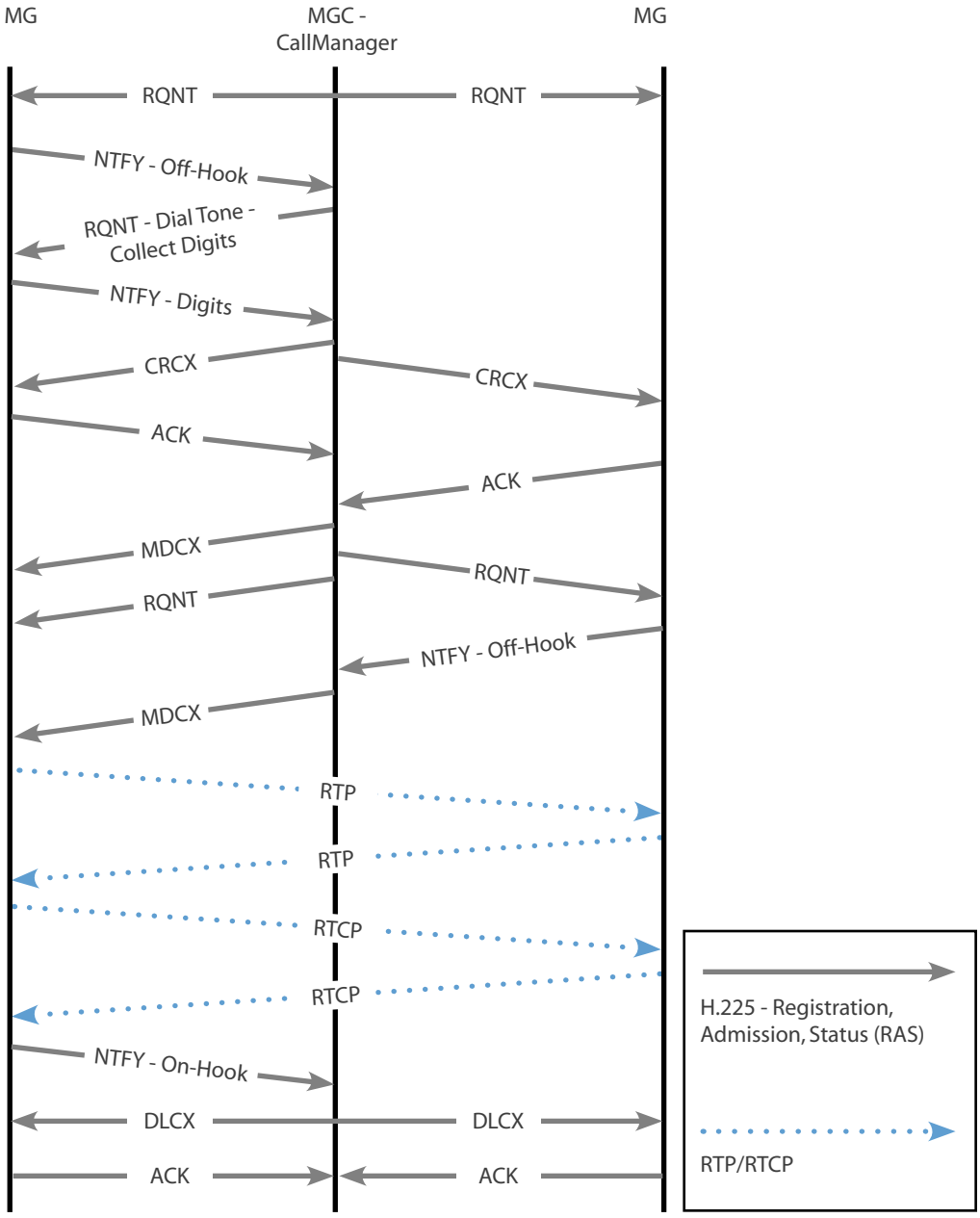


Figure 9 - MGCP Call Flow

Describe SCCP

Skinny Call Control Protocol (SCCP) is a proprietary legacy Cisco protocol which is used to connect SCCP clients to a Call Agent (CallManager). SCCP is not typically deployed in current networks and is included in this manual as a general reference. The only devices which may be seen using this protocol in modern networks are some Cisco IP phones (7900 Series), but for the most part one of the other protocols described in the document are used.

Describe SIP

Session Initiation Protocol (SIP) is similar to H.323 in that it works as a distributed model and uses several different separate protocols. SIP as opposed to H.323 was developed with the Internet in mind, and as such is text based and looks similar to Hypertext Markup Language (HTML). SIP also is addressed similarly to web pages with a URL, this URL looks like sip:far-end-user@testing.com. SIP as a protocol is used to initiate and find the target recipient. Once the recipient is found the Session Announcement Protocol (SAP) takes over by identifying what type of session is trying to be established (voice or video), this is carried over the network with the Session Description Protocol (SDP). SAP and SDP are used to create or change the parameters of a call.

SIP was build to perform four basic functions: locate users and resolve their information, negotiate capabilities, change sessions during a call, and manage the setup and teardown of the call. Since SIP was build with the Internet in mind the Domain Name Server (DNS) can be used to lookup and resolve user's SIP information. SIP also introduces an enhanced presence ability; this enables the SIP end-points or User Agents (UA) to notify other parties as to their willingness and ability to take a call. SIP also defines Watchers who have the ability to receive information as to the SIP presence of other subscribers. With SIP these UA's are also separated into two entities: the User Agent Client (UAC) and the User Agent Server (UAS). The UAC is the calling party and the UAS is the called party.

SIP Requests

SIP requests are used in combination with SIP responses. These two types of messages are how SIP communicates between the UAC and UAS's on the network.

REGISTER
A UAC sends this message to inform a SIP server of its location.

INVITE
A caller sends this message to request that another endpoint join a SIP session. This message is also sent during a call to change parameters.

ACK
This message acknowledges the final response to the INVITE.

CANCEL
This message ends a call that has not been fully established.

OPTIONS
This message queries the capabilities of the server.

BYE
This message ends a session or declines a call.

SIP Responses

The following table includes several of the SIP responses which are given, this table is not exhaustive.

Response code	Response Message	Response Description
Informational Responses		
100	Trying	Request has been received and action is being taken.
180	Ringing	Call alerting.
181	Call is being forwarded	Call being forwarded to different destination.
182	Queued	Called party temporarily unavailable, call queued.
183	Session Progress	Call Progress.
Successful Responses		
200	OK	Request has succeeded.
Redirection Responses		
300	Multiple Choices	Address resolved to multiple destinations.
301	Moved Permanently	User is no longer available at this address. Additional field exists to contain forwarding information.
302	Moved Temporarily	User is currently not available at this address. Additional field exists to contain forwarding information.
305	Use Proxy	Resource must be contacted through a proxy server. Additional field exists to contain proxy server address information.
380	Alternative Service	Call unsuccessful, alternative services may exist.
Client Failure Responses		
400	Bad Request	Request not understood.
401	Unauthorized	Request required user authentication.
403	Forbidden	Request was understood but refused to fulfill it.
404	Not Found	User does not exist at the domain specified.
405	Method Not Allowed	The method was understood but not allowed for the address identified.

table continued

406	Not Acceptable	Request is only capable of generating response entities that have content characteristics not acceptable.
407	Proxy Authentication Required	Client must authenticate with the proxy.
408	Request Timeout	Server does not respond within a suitable amount of time.
410	Gone	Resource no longer available at the server and no forwarding information is known.
480	Temporarily Unavailable	Callee's end system was contacted but is currently unavailable.
486	Busy Here	Callee's end system was contacted but not willing or able to take additional calls.
487	Request Terminated	Request terminated by a BYE or CANCEL request.
Server Failure Responses		
500	Server Internal Error	Server encountered an unexpected condition while fulfilling request.
501	Not Implemented	Server does not support the functionality required to fulfill request.
502	Bad Gateway	The server received an invalid response from a downstream server.
503	Service Unavailable	The server is unable to process the request due to temporary overloading or maintenance of the server.
504	Server Time-out	The server did not receive a timely response from an external server.
505	Version Not Supported	The server does not support the SIP protocol version used in request.
Global Failure Responses		
600	Busy Everywhere	The callee's end system was contacted successfully but the callee is busy and does not wish to take the call at this time.
603	Decline	The callee's machine was successfully contacted but the user explicitly does not wish to participate.
606	Not Acceptable	The user UA was contacted successfully but some aspect of the session description was not acceptable.

Table 2 - SIP Response Codes

Components

Proxy Server
The proxy server has the ability to perform call routing, authentication, authorization, address resolution and loop detection. The proxy servers tried to locate the called party and will relay the SIP messages along the path that it finds as well as updating the calling party (UA). Once the call setup is complete, the proxy server can be kept in the signaling path in order to see call change or termination messages.

It is important to note that a proxy server is not needed with SIP for end-to-end communications.

Redirect Server
The Redirect server has the ability to keep track of UA's which change their location either permanently or temporarily. The redirect server also has the capability to return multiple possible addresses.

Registrar Server
The registrar server is used by the UA's to register and find location information. The registrar server takes this information and places it onto the location server. Other servers in the SIP network query the registrar server for the location of called parties.

Location Server
The location server maintains the location database.

Presence Server
The presence server is responsible for gathering presence information from the presentities (UA's) and subscription information for the watchers (Other UA's).

SIP Call Flow

The following SIP call flow examples show two different situations. The first example shows the traffic between SIP gateways when one analog telephone calls another. The second shows a native SIP device calling an analog phone via a proxy server. These show a basic representation of what the SIP call flow encompasses.

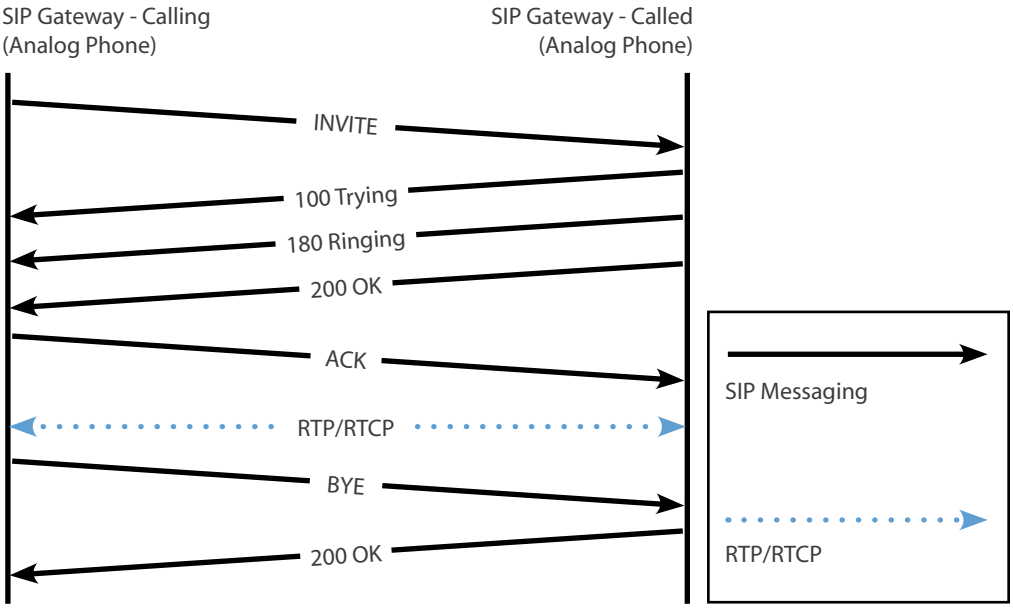


Table 3 - SIP Call Flow (Between analog phones)

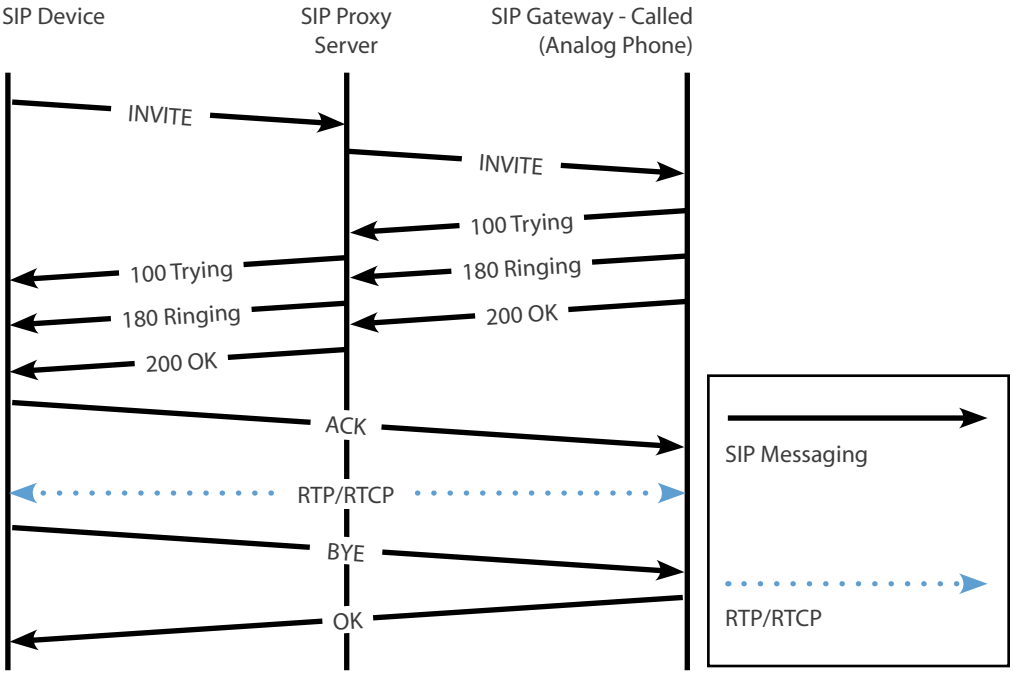


Table 4 - SIP Call Flow (Between SIP device and analog using proxy)

Identify the Appropriate Gateway Signaling Protocol for a Given Situation

The first question which must be tackled is the type of call control model you want to follow. These include the centralized call control model and the distributed call control model. Each of the three main VoIP protocols fits into one of the following call control models, these models dictate how each of the VoIP protocols communicate.

It is important to note that these are the models that the protocols were designed for, however both SIP and H.323 can be configured with call gatekeepers which can perform some amount of call control. The advantages and disadvantages that come with centralized call control come with it.

Once a selection of call control model has been made the decision changes, if a centralized mode is wanted then the MGCP protocol is utilized but if a distributed mode is a wanted then a second decision must be made between H.323 and SIP. H.323 is a specification of multiple other protocols which are used to provide a variety of services from VoIP to Video over IP. H.323 has been around longer than SIP and is more closely tied with the protocols which are used on conventional voice networks. SIP is designed around using existing Internet services like DNS as supporting protocols which make some parts of SIP implementation easier as the same equipment can be used for your data and voice networks. The main selection between H.323 and SIP is a complex one and is outside the scope of this manual; there are several excellent books detailing the advantages and disadvantages of both protocols.

Call Control Models

Distributed

The distributed call control model is used with H.323 and SIP. Distributed call control works by distributing the duties of the network over several different network components. What this means is that call setup, maintenance, routing, CAC and teardown can be controlled through several different devices. Distributed call control is easy to deploy but is harder to configure as there is no central calling authority.

Central

The central call control model is used with MGCP. Central call control works by having the call setup, maintenance, routing, CAC and teardown duties controlled by a Call Agent (CA). The CA is typically at a central location and controls a number of endpoints. The endpoints retain the digitization, encapsulation and transportation duties. Once a call has been setup between the 1st endpoint and the CA and the 2nd endpoint and the CA, the call continues between the 1st and 2nd endpoint independent of the CA until call teardown.

Describe Voice Quality Considerations

Voice quality over IP networks is a moving target as it depends greatly on a number of different factors. It must be decided based on the availability of these factors and the expected quality of the calls when determining the correct codec to use on the network. The main factors which need to be analyzed are bandwidth, delay, jitter and loss which are described in further detail below.

Bandwidth

Bandwidth is simply the amount of data that can be sent over a network at one time. Bandwidth is the easiest part of QoS to understand, to use more bandwidth then is available on the network will cause some or all of the traffic to be affected. When thinking of bandwidth in QoS terms it is typically the available bandwidth that needs to be concerned with. Available bandwidth is a measurement of the minimum bandwidth available on a path from point A to point B divided by the number of potential traffic flows. This is shown in the following figure:

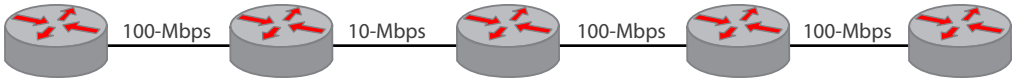


Figure 10 - Bandwidth Example

Using this figure the amount of minimum bandwidth is 10-Mbps across the whole path, if ten flows are needed then the total available bandwidth per flow is 1-Mbps.

Delay

Delay is a crucial part of QoS management. The amount of overall delay from end-to-end is very important when dealing with voice and video over networks. Optimally, the delay for a VoIP network should be less than 150 ms. There are many different things that can affect the amount of delay that is introduced from one side of a path to the other. The five main delay factors are processing delay, queuing delay, serialization delay, and propagation delay.

Processing delay is the amount of time that it takes the layer 3 devices (router or switch) to transfer a packet in one interface and out another. Many different things affect this including CPU speed, CPU utilization, total memory, available memory, and bus speed among others.

Queuing delay is the amount of time that a packet spends in the queue of a layer 3 device. Queues are used in equipment to store data when the bandwidth is currently completely utilized; this information is stored for a short time in a queue until the bandwidth opens up. The amount of time that the packet spends in these queues is the queuing delay.

Serialization delay is the amount of time that it takes for a packet to be broken down into layer 2 frames then into layer 1 electric or optical signals.

Propagation delay is the amount of time it takes for a packet to cross the physical medium.

Jitter

Jitter or delay variation is when the amount of delay changes from packet to packet which causes packets to arrive at the destination out of order as determined by the RTP time-stamping. Obviously, when dealing with a voice or video call the packets must be reassembled in the correct order or the voice or video would not make any sense. Some devices have what is called a jitter buffer which is used to mitigate small amounts of jitter by essentially creating a small queue of out of order packets and reorganizing them back into order. This can only be done correctly when the overall amount of jitter is minimal. This jitter buffer also adds additional end-to-end delay.

Loss

Loss is simple; it is the complete loss of a packet somewhere across the path from A to B. There are a number of different reasons for loss which include, output drop, input drop, overruns, ignored frames, and frame errors among others.

Output drop or *tail drop* is when a router is trying to transfer a packet from an input queue to an output queue after routing and finding the output queue to be completely full. If this happens, the packet is dropped.

Input drop is when a router tried to receive a packet but it's input queue is completely full.

Overruns are what happens when a router is trying to receive a packet but the router is so busy that it is unable to allocate buffer space for the packet. This does not mean that the buffer is full, simply that the CPU did not have the time to create it.

Ignored frames are frames which are dropped because there was no level 2 buffer space available to put them.

Frame errors occur when a frame is received but the CRC or checksum do not match which shows that the frame was corrupted in some way along the link.

Choose the Appropriate Codec for a Given Situation

The type of codec which is chosen depends on a couple of factors including the voice quality sought, the bandwidth available and delay expected between voice devices. Each of the codecs is benchmarked using a Mean Opinion Score (MOS), the MOS scale rates a codec from 1 (Bad Quality) through 5 (Great Quality). Another major factor is cost; each specific type of codec chosen uses a specific amount of Digital Signal Processing (DSP) resources.

There are three levels of complexity between codecs: low, medium and high. The following table shows some of the codecs and their complexity:

Codec Complexity	Codec
Low	G.711 (μ-law and a-law) Fax Pass-through
Medium	G.729A G.729AB G.726 Fax Relay
High	G.729 G.729B G.728 G.723.1 G.723.1A Modem Relay

Table 5 - Codec Complexity

Domain 4 - Implement a Gateway

Describe the Gateway Call Routing Process

The purpose of the gateway as described in earlier sections is to translate between one type of interface and another. Within VoIP networks this typically involves translating an analog line to a digital one; for example an FXS line to a telephone goes into a gateway to be translated onto a digital trunk line to the PSTN. When this happens the gateway must be able to understand not only the way to communicate with the telephone but also with the trunk line. When someone picks up the telephone, it is the responsibility of the gateway to see this off-hook status and initiate dialtone to the telephone, be able to receive digits and be able to correctly route the call onto the appropriate trunk line. This process of course greatly depends on the types of interfaces being used on both the sending and receiving side. Without the use of a gatekeeper the gateway must be configured with a complete picture of the VoIP network so that calls which are put through the gateway can be understood and routed. If this is not true the gateway would be just as useless as a common router without any routing information configured.

There are a number of ways to configure a gateway and a number of interfaces which are supported; with each of these interfaces are specific configured parameters which are specific to each. The following sections will go over the various common lines (or ports) which can be configured on Cisco equipment. There are also a number of different features which are common to voice networks which are also supported on this equipment; these will also be detailed in the following sections from the perspective of the gateway not using a gatekeeper.

Configure Analog Voice Ports

The configuration of analog voice ports is one of the most basic functions which are required for Cisco equipment to work in the voice world. The Cisco equipment must have a port-adapter installed that supports the type of interface that is required. The following section goes over the basic steps which are required for each different type of analog interface.

FXS

The first step which is required to configure an FXS port is to enter into voice-port configuration mode; this is done through the voice-port command. Once you have entered into voice-port configuration mode the access signaling type must be configured. On an FXS port the different options available are loop-start and ground-start. To configure the access signaling type the signal *loop-start* | *ground-start* command is used. Once the access signaling type has been configured the call progress tones must be configured, when configuring the call progress tones the tone, ring and cadence are all configured with the same command. The command that is used to configured call progress tones is cptone *locale*, the *locale* used in this command is a two-letter locale which complies with ISO 3166. By default, the us locale is used and the call-progress tone, ring and cadence would be recognizable to US phone users. The next thing to configure on an FXS port is the ring frequency, in this context this does not configure how often a ring is given (which is configured in the cptone command) but configures the frequency in Hertz that is used to ring the telephone equipment. The command to accomplish this is ring frequency *frequency*, where *frequency* can be either 25/50 Hz or 20/30 Hz depending on the equipment. The next optional command can be used to change the ring cadence which is configured by the cptone command. This cadence can be altered individually or a new pattern can be created by using the ring cadence command.

FXO

The first step which is required to configure an FXO port is to enter into voice-port configuration mode; this is done through the voice-port command. Once you have entered into voice-port configuration mode the access signaling type must be configured. On an FXO port the different options available are loop-start and ground-start. To configure the access signaling type the signal *loop-start* | *ground-start* command is used. Once the access signaling type has been configured the call progress tones must be configured, when configuring the call progress tones the tone, ring and cadence are all configured with the same command. The command that is used to configured call progress tones is cptone *locale*, the *locale* used in this command is a two-letter locale which complies with ISO 3166. By default, the us locale is used and the call-progress tone, ring and cadence would be recognizable to US phone users. The next thing to configure with an FXO port is the dial type with the options being either tone or pulse. The dial type is configured using the dial-type *pulse* | *tone* command.

E&M

The first step which is required to configure an E&M port is to enter into voice-port configuration mode; this is done through the voice-port command. Once you have entered into voice-port configuration mode the access signaling type must be configured. On an E&M port the different options available are wink-start, immediate-start or delay-start. To configure the access signaling type the signal *wink-start* | *immediate-start* | *delay-start* command is used. Once the access signaling type has been configured the call progress tones must be configured, when configuring the call progress tones the tone, ring and cadence are all configured with the same command. The command that is used to configured call progress tones is cptone *locale*, the *locale* used in this command is a two-letter locale which complies with ISO 3166. By default, the us locale is used and the call-progress tone, ring and cadence would be recognizable to US phone users. The next thing to configure on an E&M port is whether the port uses 2 wires or 4 wires; this configuration is done through the operation *2-wire* | *4-wire* command. The next thing that must be configured is the type of E&M interface used; the options are from 1 through 5. To configure the E&M type the type *1* | *2* | *3* | *4* | *5* command is used.

Configure Digital Voice Ports

Codec Complexity Configuration

Most Cisco equipment requires that the complexity of the codecs be configured before the rest of the port configuration is completed. This configuration limits the type of codec that can be used on a specific voice-card. The voice-card uses this configuration to determine the total number of calls which can be active without oversubscribing the circuit being configured. There is however some hardware which is available which allows what is called *flex* mode. In this flex mode you are allowed to configure a number of channels which exceeds the capability of the hardware if all the channels being used require the higher complexity codec. This type of configuration does require oversubscription as if a number of calls come up and all require the higher complexity codec the DSP will be unable to devote the resources to enable any more calls and actively drops new calls until the resources are freed.

In order to configure the codec complexity DS0 groups or PRI groups can be configured. This configuration requires the voice-card configuration mode. While in the voice-card configuration mode the **codec complexity** *high* | *medium* | *flex* is used to set the complexity of the codec which is to be used on this equipment.

Voice Port Controller Configuration

Clocking

One of the first things that must be understood about digital circuits is that timing is very important. As digital T1 and E1 channels work by interleaving channel data at specific intervals the source of timing on both the sending and receiving sides must be the same. If this timing is off the receiving end will interpret the data as coming from a different channel then the sender thus making the data unusable.

There are two main types of clock used on Cisco equipment, internal and line. Cisco equipment has an internal clock which can be used for network timing. While the clock is useful it is not overly accurate (from a timing perspective). For this purpose it is often best to use the clock from the line going to the telephone CO, their timing comes from a global timing source of extreme accuracy. Networking equipment can be configured to take timing from only one source at a time; if multiple line sources are available then these sources must be prioritized.

Controller Configuration
When configuring a T1/E1 card is must be configured to specify one or the other, this is done through the card type *t1* | *e1 slot* command.

Once the card type is specified then the specific controllers on the cards can be configured individually. In order to enter controller configuration mode the **controller** *t1* | *e1 slot/port* command is used. The next thing to configure is the framing used on the specific controller. For T1 circuits, the command used to specify framing is the **framing** *sf* | *esf* command. Where *sf* specifies Superframe and *esf* specifies Extended Superframe. For E1 circuits, the command used to specify framing is the **framing** *crc4* | *no-crc4* | *unframed*. Where *crc4*specifies the CRC4 mechanism in G.704 being used, *no-crc4* specifies the CRC4 mechanism in G.704 is not being used and *unframed* specifies no framing used.

The next thing to be configured is the clock source, which is done through the **clock source** *{**line** primary}* | *internal* command. In order to specify the use of the internal clock the *internal* keyword is used. When the line is to be used for timing the **line** keyword is used. If more then one line clock is specified than the *primary* keyword should be added with the **line** keyword to specify the correct line as primary.

The line code is the next thing that needs to be configured on the controller. For T1 circuits, the command **linecode** *ami* | *b8zs* is used. Where *ami* specifies Alternate Mark Inversion and *b8zs* specifies Mark Inversion using eight-zero substitution.

The final part of the controller configuration requires the creation of a DS0 channel group or a PRI channel group which is done in controller configuration mode. The channel group is used to specify a number of channels with the same configuration parameters. The creation of a DS0 channel group is done through the **ds0-group** *ds0-group-number* **timeslots** *timeslot-list* **type {e&m-delay-dial** | **e&m-immediate-start** | **e&m-wink-start** | **fxo-ground-start** | **fxo-loop-start** | **fxs-ground-start** | **fxs-loop-start}** command. This command not only specifies the type of each port in the channel group but also the type of access signaling type. The PRI channel group creates a PRI ISDN group; this requires a little more configuration then the DS0 channel groups. The first thing that must be configured is the type of ISDN switch-type which is to be used; there are a number of different switch types. The creation of the PRI group is done through the **pri-group timeslots** *timeslot-range* command. The second thing that must be done is to configure the signaling PRI channel, with T1's this channel is 24 and with E1's it is 16. This is done through the **interface** *interface interface-number* command, where T1's channel 24 would be specified with :23 and E1's channel 16 would be specified as :15 (For Example, **interface serial 0/0:23**). And finally while in interface configuration mode in order for the traffic to be routed to the DSP the **isdn incoming-voice voice** command is required.

Channel Group Voice Configuration
In order to configure the specific voice port the same configuration can be used which is referenced in the analog port configuration section including the type, cptone, ring frequency, and ring cadence among other commands.

Describe Considerations for PBX Integration

The main consideration which is very important for PBX integration is the use of the QSIG protocol which can be used as a replacement for Q.931 when connecting a trunk between two different PBX's. The support for this is done through the pri-group command using an ISDN switch type which specifies the use of QSIG (isdn switch-type primary-qsig).

Configure Dial-Peers

Basic Dial-Peer Configuration

All types of dial peers which can be configured on Cisco equipment is done through the same basic syntax. This includes the use of the dial-peer voice *number* {pots | voip} command which is issued in global configuration mode. The *number* in this case simply identifies the dial-peer on the equipment.

Assigning Voice Ports

When using POTS dial peers it is necessary to assign a specific voice port so that traffic that is matched to the POTS peer knows which port to use. This assignment is done with the port *port-id* command which is issued in dial-peer voice configuration mode. How the *port-id* is used greatly depends on the type of equipment being used and the type of port which is in use.

Specifying Session Targets

When using VoIP dial peers it is necessary to specify a session target which is simply the remote device which will be sent traffic should the dial-peer be matched. The command that is used to specify this session target is the session-target *ip-address* | *hostname* (when using VoIP) which is issued in dial-peer voice configuration mode. If the dial-peer is a VoIP peer then the target will be the IP address or hostname of the remote router; this IP address or hostname must be able to be reached through normal routing protocols for the VoIP dial peer to establish.

Destination Pattern Matching

In order to match a specific dial peer a destination pattern is used. When a POTS dial peer is used the destination pattern is used to match the traffic to a specific port, when using a VoIP dial peer it is used the traffic is matched to a specific session target. The command that is used is destination-pattern {+} *string* {T}, which is used while in dial-peer voice configuration mode. The '+' which can be used within the destination-pattern is used when the pattern being matched is a standard E.164 address. The *string* parameter can be very complex but here we will go over the most used options. The *string* itself is typically used as a match parameter against the called phone number and can be simply an exact matched phone number. However, what is typically done is a match is based on the preceding digits in a phone number. In order to match these digits a couple common matching parameters are used. The period (.) character is used to match any specific valid single character, brackets ([]) can be used to indicate a specific range and the T character can be used to indicate a variable length dial string which can end with any type of character. For example, if you are trying to route all traffic which is in the 212 area code and using the 555 prefix then you would match '212555T' or '212555….'

As a side note it should be known that the use of the **T** character as a variable match causes a dial delay, when this is used there is a 10 second dialing delay while the equipment waits for additional digits.

Digit Stripping

The way that a number is transmitted from the incoming dial peer to the outgoing dial peer depends on the type of dial peer used. If outgoing POTS dial peer is used then the static digits matched by the destination-pattern command on the peer will be stripped by default. For example, if the destination pattern was 212555…. then the POTS peer would match and strip the 212555 from the digits being transmitted and only send the other four wildcard digits. This is not true of VoIP dial peers, by default they do not strip the digits in this manner. In order to get rid of this behavior the no digit-strip command can be used while in dial-peer voice configuration mode.

If digit stripping is wanted but additional digits also need to be added to the stripped number then a couple of options have been given. These include the prefix option and the forward digits option. The prefix option allows the addition of a prefix to the stripped string before it is transmitted. So if a 9 needed to be prefixed in front of the stripped number then it would be done as in the following example. If the incoming dialed number was 2125551234 and the destination pattern was set to 212555…. and a prefix of 9 needed to be added to the beginning of the stripped number then the **prefix 9** command would be used inside the outgoing dial-peer configuration. This would result in the sent number going from 2125551234 to 91234. What the forwarding digits option provides is a way to control the number of digits which are forwarded out the outgoing dial-peer. The three options are **forward-digits** *number*, **forward-digits all**, and **forward-digits extra**. The **forward-digits** *number* command specifies a fixed number of digits which will be forwarded out the outgoing dial-peer. The **forward-digits all** command specifies that all digits will be forward which match the destination pattern (like a VoIP dial-peer). The **forward-digits extra** command specifies that all digits which come in after the matching destination pattern will also be sent.

PLAR and PLAR-OPX Connections

The are also two different types of special connections which will be reviewed. These are the Private line, automatic ringdown (PLAR) and PLAR-Off Premises eXtension (PLAR-OPX). A PLAR connection which is also commonly called a ringdown line is used when a port is configured to only connect to one extension, when the line goes off-hook it will automatically ring a specific extension. To configure this, the connection plar *phone-number* command is used while in voice-port configuration mode. This command configures the device of the port to automatically call the configured *phone-number* as soon as it goes off-hook. A PLAR-OPX connection is used when a voice device is off premises and you want to configure a line in that location which is treated as if it is local, including inside dial plans and call routing. To configure this the connection plar-opx *phone-number* command is used while in voice-port configuration mode. This configuration works by routing all calls going to that specific port off to the off premises location.

Configure Hunt Groups and Trunk Groups

Hunt Groups

Hunt groups are used when more than one dial-peer matches a specific destination pattern. By default, the behavior which happens when this occurs is for the dial-peer with the longest destination pattern match to be chosen first, if the destination patterns are the same then the one with the highest priority as configured with the preference command (when using the preference command the lower the number the higher the priority) and then if the preferences are all the same then the dial-peer is chosen at random. If you want to stop or modify this hunting process, two options are available. Either disable or modify the selection criteria with the dial-peer hunt command (no dial-peer hunt disables hunting) or by using the huntstop command on a specific dial-peer. The huntstop command instructs the equipment to stop hunting if the configured peer is busy or unavailable.

Trunk Groups

The purpose of a Trunk Group is to group a number of ports together and configure their options in one group. Put simpler a trunk group is a hunt group of trunks. When configuring a Trunk Group there are four basic steps which are required: configuring the Trunk Group and assigning parameters to the trunk group, assigning controllers to a specific trunk group, assigning voice-ports to a specific trunk group and configuring a dial-peer to use a specific trunk group to route traffic.

In order to create and configure a trunk group the **trunk group** *name* command is used while in global configuration mode. There are a various number of commands which can be used after creating the trunk group with the hunt scheme being one of the most used commands. Once the trunk group has been created then the controller and the ports can be assigned to the specific trunk group. This is done through the **trunk-group** *name* command which is used while in controller and voice-port configuration modes, respectively. Once this is complete the dial-peers can be configured to use specific trunk groups for outgoing calls; trunk groups are only used on outgoing calls. The configuration on the dial-peers is done through the **trunkgroup** *name priority* command, the lower the priority the higher the preference.

Configure Digit Manipulation

Number Expansion

Number expansion is typically used for what the name suggests, expanding a number. An example of this is used in many companies and universities today; inside the organization the numbers are contacted via an extension. However, this number can not be given as a called number to an outside telephony device. When you call from one part of the organization to another through a public telephony system which requires the whole number for call routing then number expansion can be very useful. This number extension is done through the num-exp *extension-number expanded-number* command, and is used while in global configuration mode. For example, if there was a group of extensions which were all four digits long and all started with 1, the num-exp command could be configured as num-exp 1… 5551….This example would then take any number matched through the destination pattern command that also matched this command and expand it for you.

The second reason to use the number expansion feature is if a number or extension needs to be forwarded to a completely different number. For example, if the 1234 extension needed to be forwarded to 5554321 then the **num-exp** 1234 5554321 command could be used.

Translation Rules

Translation rules can be used to change a number before the call is matched to an inbound dial peer or before a call is forwarded by the outbound dial peer. To configure the translation rules the voice translation-rule number command is used while in global configuration mode. While in translation–rule configuration mode you enter a number of rules with the rule precedence match-pattern replace-pattern command.

The matching patterns use similar rules to that of the destination-pattern command. The translation rule is then assigned to the specific dial-peer with either the translate-outgoing {calling-number | called-number} number command.

Configure Calling Privileges

Calling privileges are configured on Cisco equipment through the use of Class of Restrictions (COR). COR is configured in a couple of steps. First the name of the COR is configured, this must be done while in dial-peer COR custom configuration mode. In order to get into this mode the dial-peer cor custom command is used while in global configuration mode. Once in dial-peer COR custom configuration mode the different COR classes are configured using the name *class-name* command. Once the different classes have been configured then a number of COR lists need to be configured, these lists are to be configured for incoming and outgoing dial-peers. The basic idea is that the incoming dial-peers are able to route calls to outgoing dial-peers which are in the same COR class. In order to configure these lists you must be in dial-peer COT list configuration mode, in order to get into this mode you use the dial-peer cor list *list-name* command.

Once in this mode you can use the member *class-name* to link the list to the COR classes already configured with the name command. Once all the different lists are configured then the list must be linked to specific dial-peers, this is done through two commands: corlist incoming *list-name* and corlist outgoing *list-name*.

Now as we know this topic is a bit confusing because of this the following example shows a basic configuration.

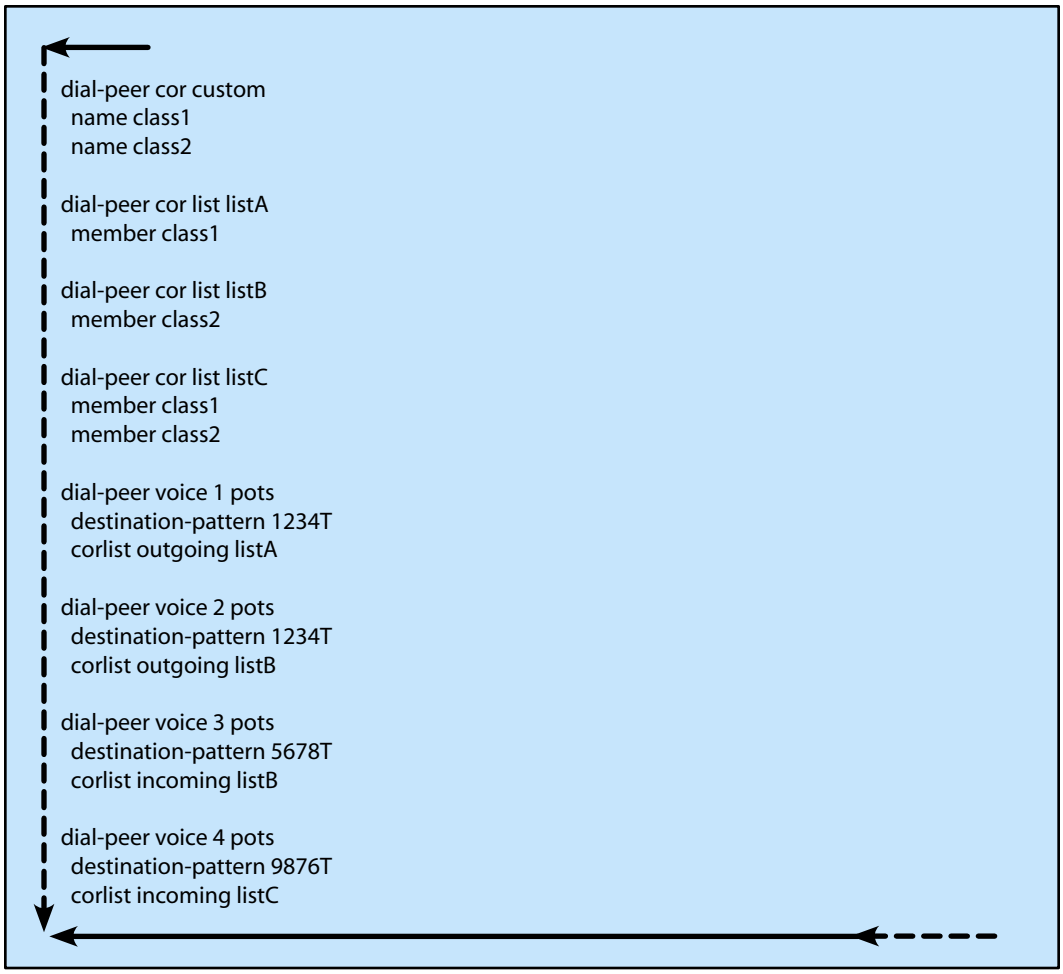


Figure 11 - COR example

In the above configuration, calls coming in POTS dial-peer 3 will be able to make calls to POTS dial-peer 2 only because its incoming COR list is limited to COR list class2. Calls coming in POTS dial-peer 4 will be able to make calls through both POTS dial-peer 1 and 2 because its incoming COR list is linked to both COR list class1 and class2.

Verify Dial-Plan Implementation

The first command that can be used to verify the dial-plan is the show dialplan number *dial-string* command. This command when used displays the matching outgoing dial-peer which is used for a specific *dial-string*; this can be used to make sure that the configuration correctly matches to the outgoing dial-peer expected.

The second command to use actually requires two commands, the **debug voice dialpeer all** command and the **csim start** *number* command. The **csim start** command simulates a call being placed to the *number* specified and the **debug voice dialpeer all** command enables the display of the debug information coming from the call being placed. This is a very good way of testing how a call to a specific number will be routed through the equipment.

Implement Fax and Modem Support on a Gateway

Faxing Support

When faxing over a VoIP network there are two main ways to perform this, fax pass-through and fax relay. Fax pass-through works by treating the call as a typical voice call and sending the fax data in-band using RTP packets. The only difference is that if the gateway detects a fax it will force the G.711 codec (64 kbps). Fax relay works quite a bit differently, and within Fax relay there are two different implementations: Cisco Fax Relay and T.38 Standards based fax relay. Both Cisco fax relay and the T.38 fax relay work the same coming into the gateway. Both take the T.30 fax signal created by the fax machine and demodulate the signal (back to its original form, before it was prepared for telephone transfer). Where the two methods differ is that Cisco fax relay keeps the T.30 fax signal and transports it over RTP to the destination gateway, the destination gateway then modulates the signal and returns it to the remote fax machine. With T.38 fax relay the signal is converted from a T.30 signal into a T.38 signal (Internet Fax Protocol (IFP)). At the destination gateway the T.38 signal is converted back to a T.30 signal and modulates the signal and returns it to the remote fax machine.

Cisco Fax Relay

There are two ways to configure Cisco fax relay, one is done per dial-peer and the other sets the relay method globally on the networking equipment. The command to enable Cisco fax relay on a dial-peer is **fax protocol cisco** which is entered in dial-peer voice configuration mode. To enable Cisco fax relay on all dial-peers the **fax protocol cisco** command is used while in voice service configuration mode.

T.38 Fax Relay

T.38 fax relay is configured different depending on the type of VoIP signaling used. If using MGCP the no mgcp fax t38 inhibit command is used while in global configuration mode, by default MGCP T.38 fax relay is enabled. If using SIP or H.323 the configuration is done in one of two ways, either per dial-peer or globally on the networking equipment. The command to enable T.38 fax relay on a dial-peer is fax protocol t38 which is entered in dial-peer voice configuration mode. To enable T.38 fax relay on all dial-peers the fax protocol t38 command is used while in voice service configuration mode.

Fax Passthrough

Fax passthrough is configured differently depending on the type of VoIP signaling used. If using MGCP the mgcp modem passthrough voip mode nse command is used while in global configuration mode, this command uses the NSE method of changing modem speeds. There is also a Cisco method which is enabled by using the mgcp modem passthrough voip mode cisco command. If using SIP or H.323 the configuration is done in one of two ways, either per dial-peer or globally on the networking equipment. The command to enable fax passthrough on a dial-peer is fax protocol passthrough { g711ulaw | g711alaw} which is entered in dial-peer voice configuration mode. To enable fax passthrough on all dial-peers the fax protocol passthrough { g711ulaw | g711alaw} command is used while in voice service configuration mode.

Modem Support

Cisco Modem Relay

Cisco modem relay is possible if specific conditions exist as well as the proper configuration. The requirements include Cisco originating and terminating gateways, both modems must support V.34 or V.90 and must use V.42bis compression; both must also have error correction enabled. If these conditions are met then modem configuration can be completed. If using MGCP, the command to use is mgcp modem relay voip mode nse while in global configuration mode. If using SIP or H.323 the configuration is done in one of two ways, either per dial-peer or globally on the networking equipment. The command to enable mode relay on a dial-peer is modem relay nse codec {g711alaw | g711ulaw} which is entered in dial-peer voice configuration mode. To enable modem relay on all dial-peers the modem relay nse codec {g711alaw | g711ulaw} command is used while in voice service configuration mode.

Modem Passthrough

Modem passthrough is configured in the same way that is detailed in the Fax Passthrough section.

Configure a Gateway to Provide DTMF Support

While the methods are similar between the call signaling protocols they are not the same, because of this each call signaling protocol is detailed separately.

MGCP

When using MGCP there are four ways to relay DTMF: Cisco Proprietary, RTP-NSE (Named Service Event), Named Telephony Event (NTE) and out-of-band. The command syntax for all variants is similar, one of the common options between them is the option for the relay to be used for all calls or for only low-bandwidth codecs, these are configured with the all and low-bit-rate. The Cisco Proprietary method sends the DTMF signals in the same RTP stream as the voice and is configured with the mgcp dtmf-relay voip codec {all | low-bit-rate} mode cisco command. The RTP-NTE method is standards based on RFC 2833, this method is also used inside the RTP stream and is encapsulated in an NTE packet. The RTP-NSE method is configured using the mgcp dtmf-relay voip codec {all | low-bit-rate} mode nse. There is also two different MGCP specific NTE variants, gateway controlled (NTE-GW) and call agent controlled (NTE-CA). These are configured with the mgcp dtmf-relay voip codec {all | low-bit-rate} mode nte-gw and mgcp dtmf-relay voip codec {all | low-bit-rate} mode nte-ca commands, respectively. The final method sends the DTMF signals out-of-band through a control channel to CallManager, this is configured with the mgcp dtmf-relay voip codec {all | low-bit-rate} mode out-of-band command.

H.323

With H.323 there are also four ways to relay DTMF: Cisco Proprietary, RTP-NTE, H.245 Alphanumeric and H.245 Signal. The Cisco proprietary method works the same as described above but is configured differently for H.323. In order to configured the use of the Cisco proprietary DTMF relay method with H.323 the dtmf-relay cisco-rtp command is used while in dial-peer voice configuration mode. The RTP-NTE method also works the same as with MGCP but requires the use of the dtmf-relay rtp-nte command while in dial-peer voice configuration mode. The two other methods are specific to H.323. The H.245 alphanumeric method uses the H.245 signaling channel to transmit DTMF, it does this by transmitting the ASCII character which represents the DTMF code to the receiving gateway. This method is configured using the dtmf-relay h245-alphanumeric command while in dial-peer voice configuration mode. The problem with H.245 alphanumeric mode is that it does not sent along the duration of the DTMF signal it simply assumes the signals last 500 ms, this can be problematic using some phone systems. To remedy this problem there is another type of H.245 DTMF relay which sends the DTMF signal the same way but also includes the length of the signal; this is called the H.245 Signal. This is configured using the dtmf-relay h.245 signal command while in dial-peer voice configuration mode.

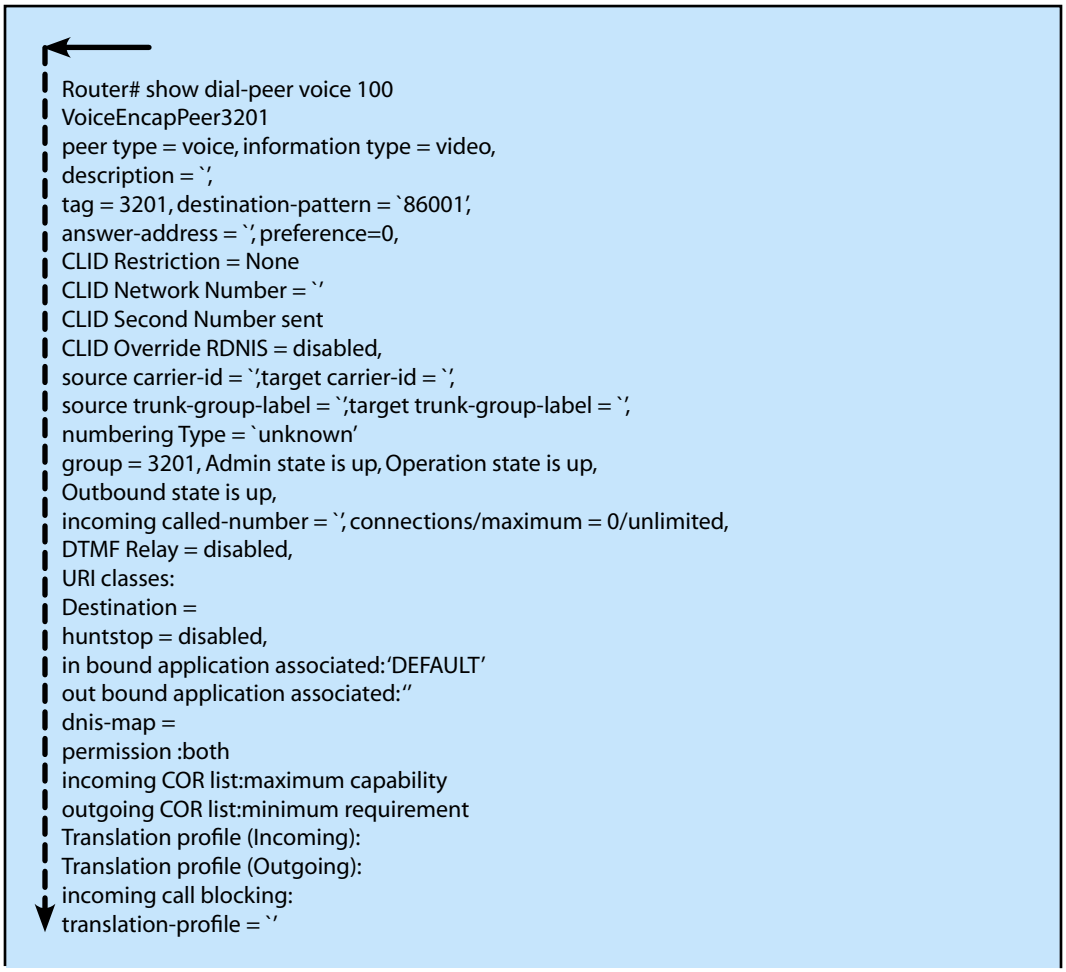
SIP

With SIP there are three main ways to relay DTMF: RTP-NTE, SIP INFO and SIP NOTIFY. With SIP RTP-NTE works the same and is configured the same as with H.323 using the dtmf-relay rtp-nte command. The SIP INFO method utilizes SIP INFO messages to transport the DTMF signals from source to receiver; SIP INFO is always enabled and can not be enabled, disabled or configured. The SIP NOTIFY method provides the services offered with the SIP INFO method (and is used in conjunction) but also provides a way to notify the application of DTMF events through the NOTIFY message type. In order to configure the SIP NOTIFY DTMF method use the dtmf-relay sip-notify command while in dial-peer voice configuration mode. You can also use the RTP-NTE and SIP NOTIFY methods together with the SIP NOTIFY as primary, this way if the receiving gateway does not support SIP NOTIFY then the DTMF signals work through RTP-NTE.

Troubleshooting

show dial-peer voice

This command is used to display information about configured dial-peers.



```
Router# show dial-peer voice 100
VoiceEncapPeer3201
peer type = voice, information type = video,
description = `',
tag = 3201, destination-pattern = `86001',
answer-address = `', preference=0,
CLID Restriction = None
CLID Network Number = `'
CLID Second Number sent
CLID Override RDNIS = disabled,
source carrier-id = `',target carrier-id = `',
source trunk-group-label = `',target trunk-group-label = `',
numbering Type = `unknown'
group = 3201, Admin state is up, Operation state is up,
Outbound state is up,
incoming called-number = `', connections/maximum = 0/unlimited,
DTMF Relay = disabled,
URI classes:
Destination =
huntstop = disabled,
in bound application associated:'DEFAULT'
out bound application associated:''
dnis-map =
permission :both
incoming COR list:maximum capability
outgoing COR list:minimum requirement
Translation profile (Incoming):
Translation profile (Outgoing):
incoming call blocking:
translation-profile = `'
```

continued below

```
disconnect-cause = `no-service'
advertise 0x40 capacity_update_timer 25 addrFamily 4 oldAddrFamily 4
type = pots, prefix = `',
forward-digits 4
session-target = `', voice-port = `2/0:23',
direct-inward-dial = enabled,
digit_strip = enabled,
register E.164 number with H323 GK and/or SIP Registrar = TRUE
fax rate = system, payload size = 20 bytes
supported-language = ''
preemption level = `routine'
bandwidth:
maximum = 384 KBits/sec, minimum = 64 KBits/sec
voice class called-number:
inbound = `', outbound = `1'
Time elapsed since last clearing of voice call statistics never
Connect Time = 0, Charged Units = 0,
Successful Calls = 0, Failed Calls = 0, Incomplete Calls = 0
Accepted Calls = 0, Refused Calls = 0,
Last Disconnect Cause is "",
Last Disconnect Text is "",
Last Setup Time = 0.
```

show call application gateway-level

This command is used to display statistics for gateway-level voice application instances.

```
Router# show call application active gateway-level
Gateway level statistics for active application sessions:
Sessions w/ stats: 1
Currently connected incoming PSTN legs: 1
Currently connected outgoing PSTN legs: 0
Currently connected incoming VoIP legs: 0
Currently connected outgoing VoIP legs: 0
Placecalls in transit: 0
Handouts in transit: 0
Pending ASNL subscriptions: 0
Pending ASNL unsubscriptions: 0
Prompts playing (non-TTS): 0
Recordings: 0
TTS prompts playing: 0
```

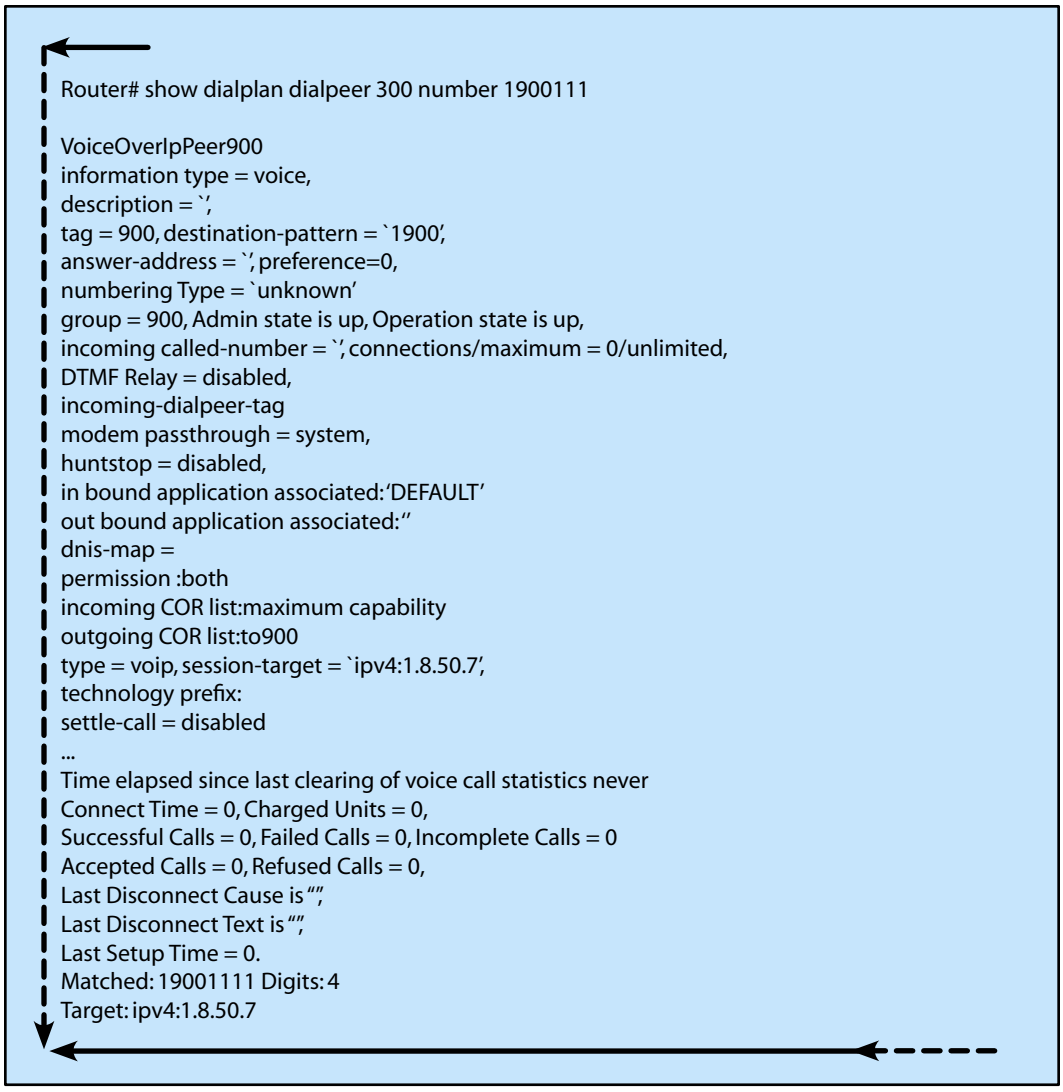
show controllers timeslots

This command is used to display information about configured timeslots on the controllers.

```
Router# show controllers timeslots
T1 1 is up:
Loopback: NONE
DS0 Type        Modem   <->             Service         Channel         Rx
Tx
                                State           State           A B C D   A B C D
------------------------------------------------------------------------------------
1 cas-modem     1       in      insvc           connected                 1 1 1 1
1 1 1 1
2 cas           -       -       insvc           idle            0 0 0 0   0 0 0 0
3 cas           -       -       insvc           idle            0 0 0 0   0 0 0 0
4 cas           -       -       insvc           idle            0 0 0 0   0 0 0 0
5 cas           -       -       insvc           idle            0 0 0 0   0 0 0 0
6 cas           -       -       insvc           idle            0 0 0 0   0 0 0 0
7 cas           -       -       insvc           idle            0 0 0 0   0 0 0 0
8 cas           -       -       insvc           idle            0 0 0 0   0 0 0 0
9 cas           -       -       insvc           idle            0 0 0 0   0 0 0 0
10 cas          -       -       maint           static-bo       0 0 0 0   1 1 1 1
11 cas          -       -       maint           static-bo       0 0 0 0   1 1 1 1
12 cas          -       -       maint           static-bo       0 0 0 0   1 1 1 1
13 cas          -       -       maint           static-bo       0 0 0 0   1 1 1 1
14 cas          -       -       maint           static-bo       0 0 0 0   1 1 1 1
15 cas          -       -       maint           static-bo       0 0 0 0   1 1 1 1
16 cas          -       -       maint           static-bo       0 0 0 0   1 1 1 1
17 cas          -       -       maint           static-bo       0 0 0 0   1 1 1 1
18 cas          -       -       maint           static-bo       0 0 0 0   1 1 1 1
19 cas          -       -       maint           dynamic-bo      0 0 0 0   1 1 1 1
20 cas          -       -       maint           dynamic-bo      0 0 0 0   1 1 1 1
21 cas          -       -       maint           dynamic-bo      0 0 0 0   1 1 1 1
22 unused
23 unused
24 unused
```

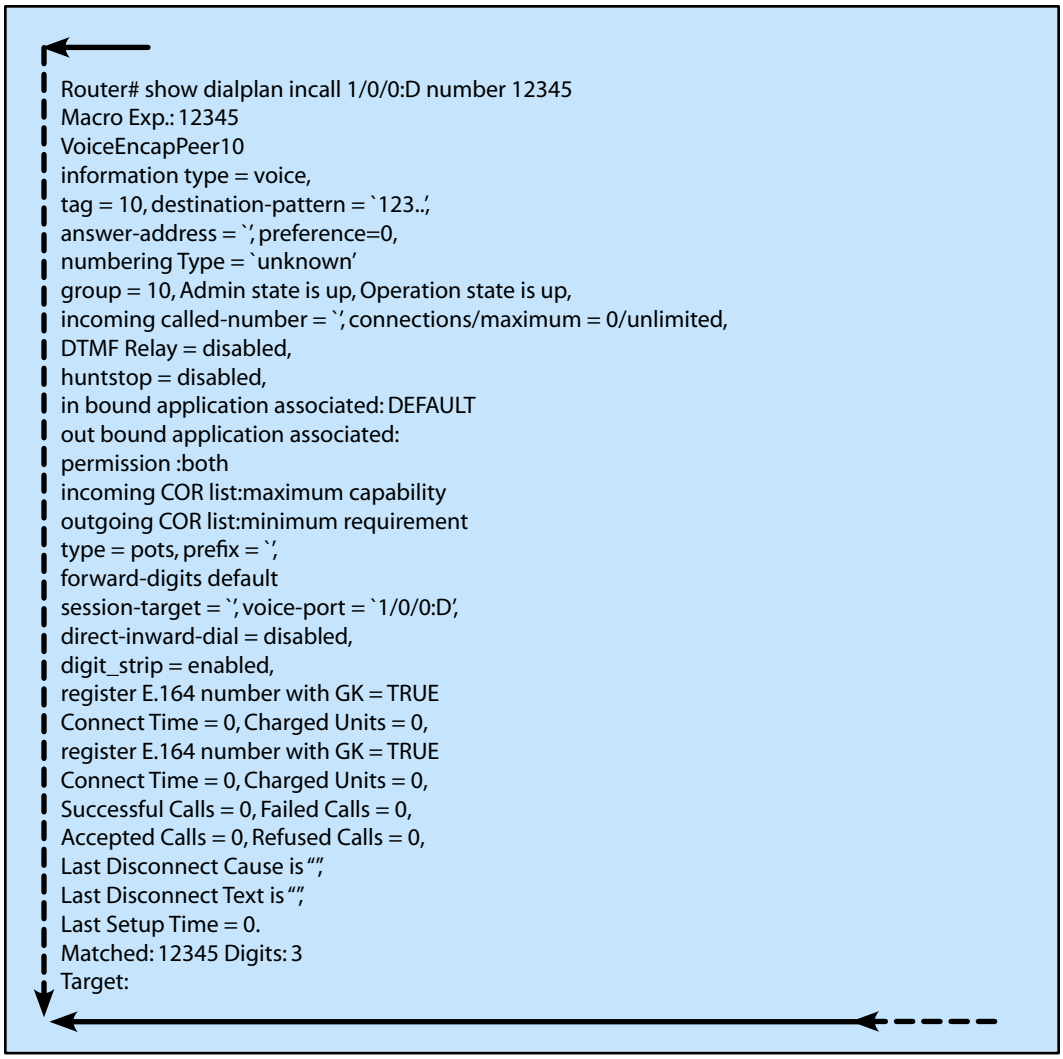
show dialplan dialpeer

This command is used to display information about configured outbound dial-peers which matched an incoming dial-peer based on COR criteria.



show dialplan incall

This command is used to display information about configured POTS dial-peer which matches based on a specific calling number or voice port.



show dialplan number

This command is used to display information about which outbound dial-peer is matched based on a specific dialed-number.

```
Router# show dialplan number 1001

Macro Exp.: 1001

VoiceEncapPeer1003
        information type = voice,
        tag = 1003, destination-pattern = `1001',
        answer-address = `', preference=0,
        numbering Type = `unknown'
        group = 1003, Admin state is up, Operation state is up,
        incoming called-number = `', connections/maximum = 0/unlimited,
        DTMF Relay = disabled,
        huntstop = enabled,
        type = pots, prefix = `',
        forward-digits default
        session-target = `', voice-port = `1/1',
        direct-inward-dial = disabled,
        Connect Time = 0, Charged Units = 0,
        Successful Calls = 0, Failed Calls = 0, Incomplete Calls = 0
        Accepted Calls = 0, Refused Calls = 0,
        Last Disconnect Cause is "",
        Last Disconnect Text is "",
        Last Setup Time = 0.
Matched: 1001 Digits: 4
Target:

VoiceEncapPeer1004
        information type = voice,
        tag = 1004, destination-pattern = `1001',
        answer-address = `', preference=0,
        numbering Type = `unknown'
        group = 1004, Admin state is up, Operation state is up,
        ...
Matched: 1001 Digits: 4
Target:

VoiceEncapPeer1002
        information type = voice,
        tag = 1002, destination-pattern = `1001',
        answer-address = `', preference=0,
        numbering Type = `unknown'
        group = 1002, Admin state is up, Operation state is up,
        ...
```

continued below

```
Matched: 1001 Digits: 4
Target:

VoiceEncapPeer1001
        information type = voice,
        tag = 1001, destination-pattern = `1001',
        answer-address = `', preference=0,
        numbering Type = `unknown'
        group = 1001, Admin state is up, Operation state is up,
        ...
Matched: 1001 Digits: 4
Target:
```

show gateway

This command is used to display information about the current status of the gateway.

```
Router# show gateway
Gateway gateway1 is not registered to any gatekeeper
Gateway alias list
H323-ID gateway1
H323 resource thresholding is Enabled but NOT Active
H323 resource threshold values:
DSP: Low threshold 60, High threshold 70
DS0: Low threshold 60, High threshold 70
```

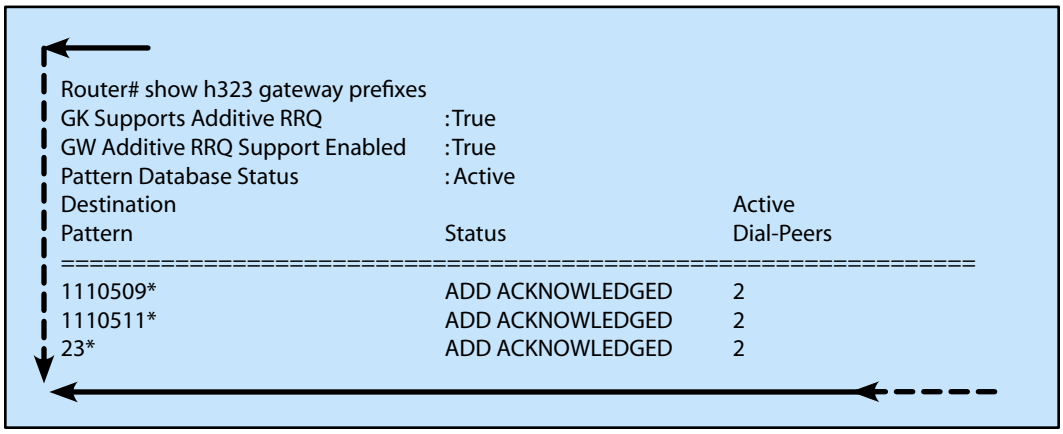
show h323 gateway

This command is used to display information statistics about the H.323 gateway messages.

```
Router# show h323 gateway

H.323 STATISTICS AT 01:45:55

H.225 REQUESTS      SENT      RECEIVED          FAILED
Setup               0         5477              0
Setup confirm       5424      0                 0
Alert               2734      0                 0
Progress            2701      0                 0
Call proceeding     5477      0                 0
Notify              0         0                 0
Info                0         0                 0
User Info           0         0                 0
Facility            2732      0                 0
Release             5198      5313              241
Reject              0         0                 0
Passthrough         0         0                 0

H225 establish timeout 0
RAS failed          0
H245 failed         0

RAS MESSAGE         REQUESTS SENT     CONFIRMS RCVD     REJECTS RCVD
GK Discovery        grq 0             gcf 0             grj 0
Registration        rrq 130           rcf 130           rrj 0
Admission           arq 5477          acf 5477          arj 0
Bandwidth           brq 0             bcf 0             brj 0
Disengage           drq 5439          dcf 5439          drj 0
Unregister          urq 0             ucf 0             urj 0
Resource Avail      rai 0             rac 0
Req In Progress     rip 0

RAS MESSAGE         REQUESTS RCVD     CONFIRMS SENT     REJECTS SENT
GK Discovery        grq 0             gcf 0             grj 0
Registration        rrq 0             rcf 0             rrj 0
Admission           arq 0             acf 0             arj 0
Bandwidth           brq 0             bcf 0             brj 0
Disengage           drq 0             dcf 0             drj 0
Unregister          urq 0             ucf 0             urj 0
Resource Avail      rai 0             rac 0
Req In Progress     rip 0

DISC CAUSE CODE     FROM OTHER PEER          FROM H323 PEER
16 normal call clearing   66                 5325
31 normal, unspecified    1                  0
34 no circuit             31                 0
41 temporary failure      3                  0
44 no requested circuit   13                 0
```

show h323 gateway prefixes

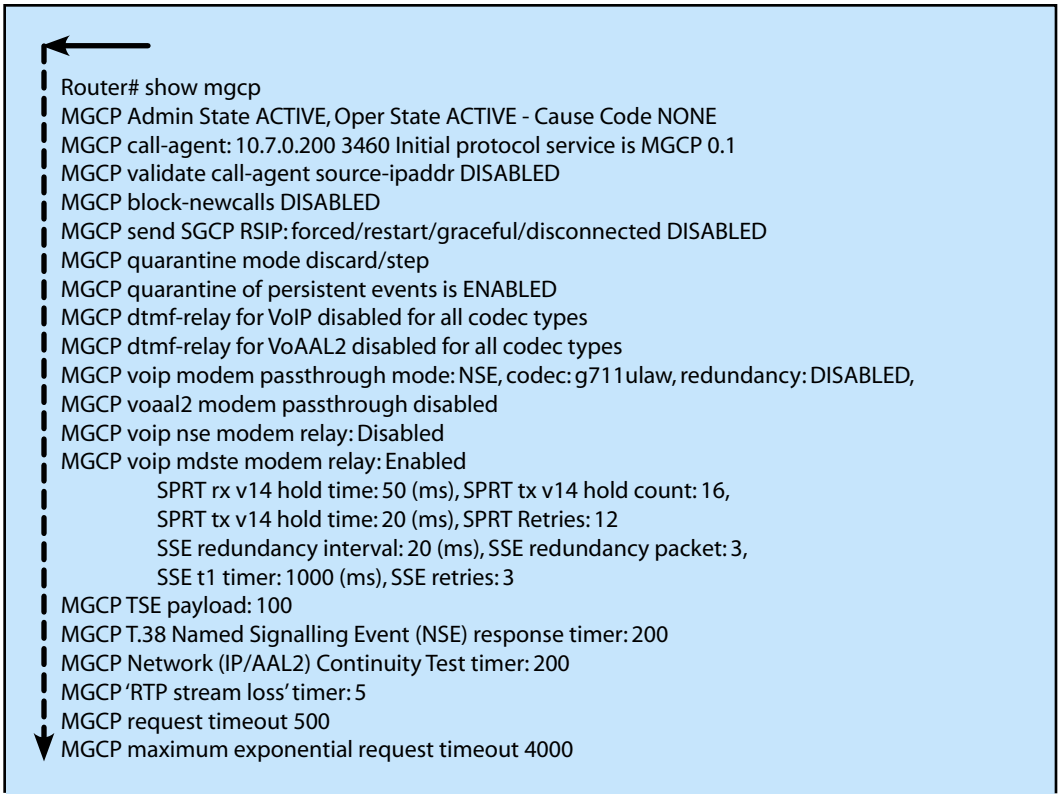
This command is used to display information about configured destination-patterns.



```
Router# show h323 gateway prefixes
GK Supports Additive RRQ              :True
GW Additive RRQ Support Enabled       :True
Pattern Database Status               :Active
Destination                                                 Active
Pattern                               Status                Dial-Peers
============================================================================
1110509*                              ADD ACKNOWLEDGED      2
1110511*                              ADD ACKNOWLEDGED      2
23*                                   ADD ACKNOWLEDGED      2
```

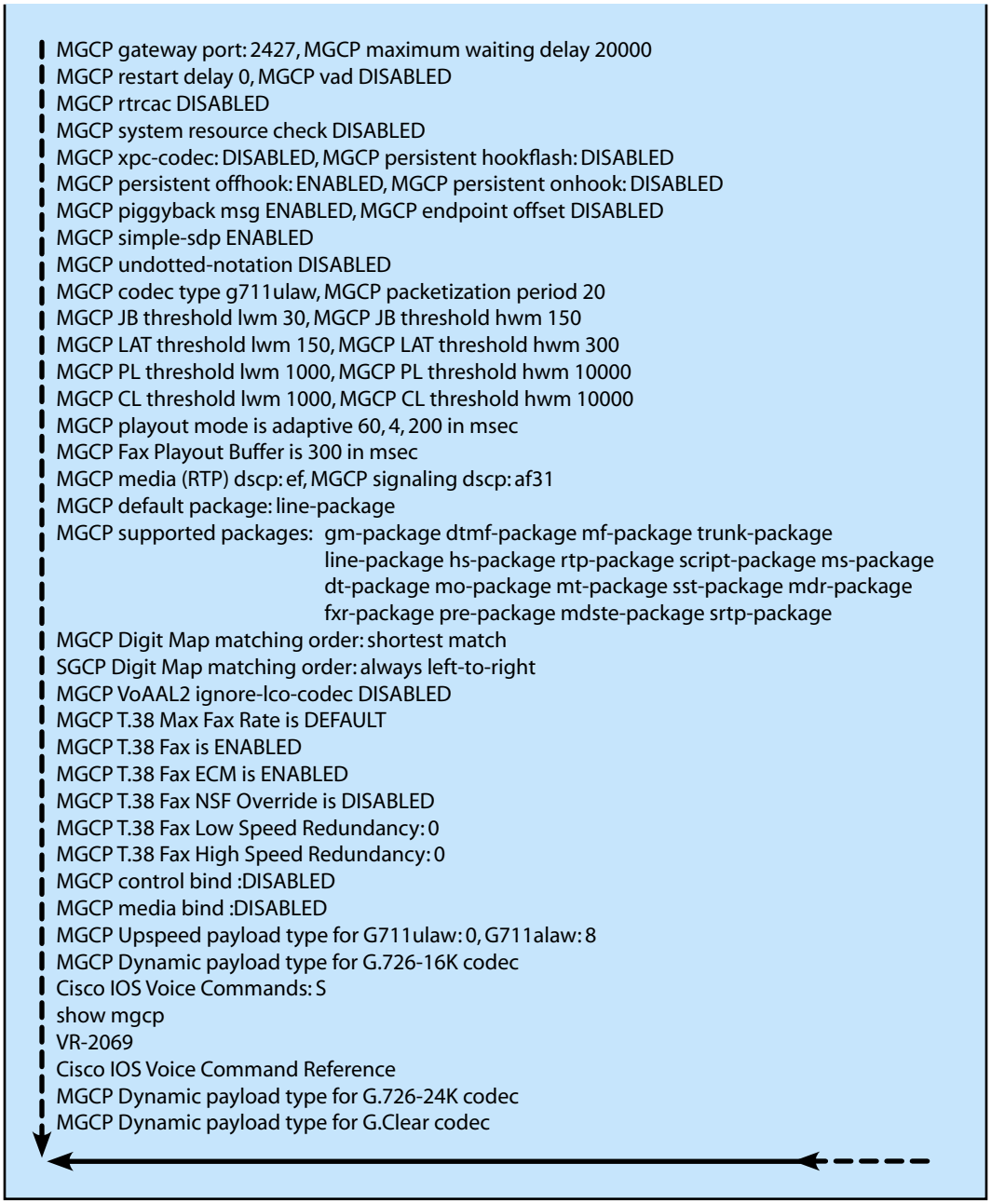
show mgcp

This command is used to display information about configured MGCP parameters.



```
Router# show mgcp
MGCP Admin State ACTIVE, Oper State ACTIVE - Cause Code NONE
MGCP call-agent: 10.7.0.200 3460 Initial protocol service is MGCP 0.1
MGCP validate call-agent source-ipaddr DISABLED
MGCP block-newcalls DISABLED
MGCP send SGCP RSIP: forced/restart/graceful/disconnected DISABLED
MGCP quarantine mode discard/step
MGCP quarantine of persistent events is ENABLED
MGCP dtmf-relay for VoIP disabled for all codec types
MGCP dtmf-relay for VoAAL2 disabled for all codec types
MGCP voip modem passthrough mode: NSE, codec: g711ulaw, redundancy: DISABLED,
MGCP voaal2 modem passthrough disabled
MGCP voip nse modem relay: Disabled
MGCP voip mdste modem relay: Enabled
        SPRT rx v14 hold time: 50 (ms), SPRT tx v14 hold count: 16,
        SPRT tx v14 hold time: 20 (ms), SPRT Retries: 12
        SSE redundancy interval: 20 (ms), SSE redundancy packet: 3,
        SSE t1 timer: 1000 (ms), SSE retries: 3
MGCP TSE payload: 100
MGCP T.38 Named Signalling Event (NSE) response timer: 200
MGCP Network (IP/AAL2) Continuity Test timer: 200
MGCP 'RTP stream loss' timer: 5
MGCP request timeout 500
MGCP maximum exponential request timeout 4000
```

continued below


MGCP gateway port: 2427, MGCP maximum waiting delay 20000
MGCP restart delay 0, MGCP vad DISABLED
MGCP rtrcac DISABLED
MGCP system resource check DISABLED
MGCP xpc-codec: DISABLED, MGCP persistent hookflash: DISABLED
MGCP persistent offhook: ENABLED, MGCP persistent onhook: DISABLED
MGCP piggyback msg ENABLED, MGCP endpoint offset DISABLED
MGCP simple-sdp ENABLED
MGCP undotted-notation DISABLED
MGCP codec type g711ulaw, MGCP packetization period 20
MGCP JB threshold lwm 30, MGCP JB threshold hwm 150
MGCP LAT threshold lwm 150, MGCP LAT threshold hwm 300
MGCP PL threshold lwm 1000, MGCP PL threshold hwm 10000
MGCP CL threshold lwm 1000, MGCP CL threshold hwm 10000
MGCP playout mode is adaptive 60, 4, 200 in msec
MGCP Fax Playout Buffer is 300 in msec
MGCP media (RTP) dscp: ef, MGCP signaling dscp: af31
MGCP default package: line-package
MGCP supported packages: gm-package dtmf-package mf-package trunk-package
line-package hs-package rtp-package script-package ms-package
dt-package mo-package mt-package sst-package mdr-package
fxr-package pre-package mdste-package srtp-package
MGCP Digit Map matching order: shortest match
SGCP Digit Map matching order: always left-to-right
MGCP VoAAL2 ignore-lco-codec DISABLED
MGCP T.38 Max Fax Rate is DEFAULT
MGCP T.38 Fax is ENABLED
MGCP T.38 Fax ECM is ENABLED
MGCP T.38 Fax NSF Override is DISABLED
MGCP T.38 Fax Low Speed Redundancy: 0
MGCP T.38 Fax High Speed Redundancy: 0
MGCP control bind :DISABLED
MGCP media bind :DISABLED
MGCP Upspeed payload type for G711ulaw: 0, G711alaw: 8
MGCP Dynamic payload type for G.726-16K codec
Cisco IOS Voice Commands: S
show mgcp
VR-2069
Cisco IOS Voice Command Reference
MGCP Dynamic payload type for G.726-24K codec
MGCP Dynamic payload type for G.Clear codec

show mgcp connection

This command is used to display information about configured MGCP connections.

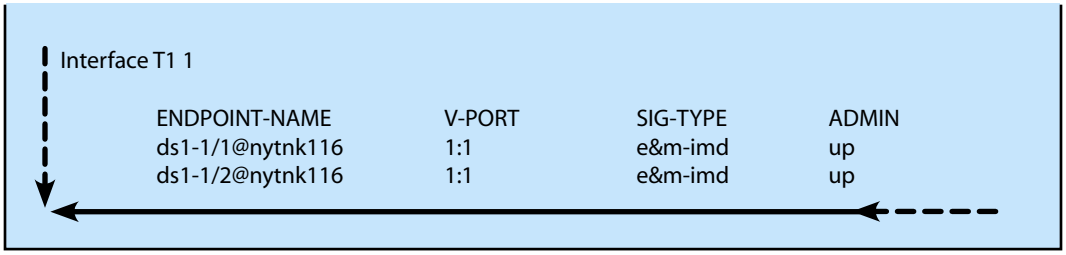
```
Router# show mgcp connection
Endpoint Call_ID(C) Conn_ID(I) (P)ort (M)ode (S)tate (C)odec (E)vent[SIFL] (R)esult[EA]
1. S0/DS1-0/1 C=103,23,24 I=0x8 P=16586,16634 M=3 S=4,4 C=5 E=2,0,0,2 R=0,0
2. S0/DS1-0/2 C=103,25,26 I=0x9 P=16634,16586 M=3 S=4,4 C=5 E=0,0,0,0 R=0,0
3. S0/DS1-0/3 C=101,15,16 I=0x4 P=16506,16544 M=3 S=4,4 C=5 E=2,0,0,2 R=0,0
4. S0/DS1-0/4 C=101,17,18 I=0x5 P=16544,16506 M=3 S=4,4 C=5 E=0,0,0,0 R=0,0
5. S0/DS1-0/5 C=102,19,20 I=0,6 P=16572,16600 M=3 S=4,4 C=5 E=2,0,0,2 R=0,0
6. S0/DS1-0/6 C=102,21,22 I=0x7 P=16600,16572 M=3 S=4,4 C=5 E=0,0,0,0 R=0,0
Total number of active calls 6
```

show mgcp endpoint

This command is used to display information about the configured MGCP controlled endpoints.

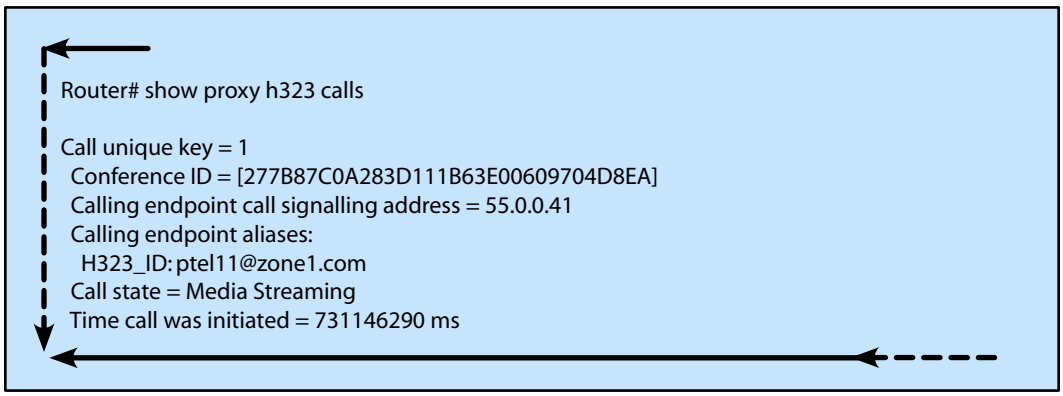
```
Router# show mgcp endpoint
        ENDPOINT-NAME          V-PORT      SIG-TYPE        ADMIN
        ds1-0/1@nytnk116       0:1         fxs-gs          up
        ds1-0/2@nytnk116       0:1         fxs-gs          up
        ds1-0/3@nytnk116       0:1         fxs-gs          up
        ds1-0/4@nytnk116       0:1         fxs-gs          up
        ds1-0/5@nytnk116       0:1         fxs-gs          up
        ds1-0/6@nytnk116       0:1         fxs-gs          up
        ds1-0/7@nytnk116       0:1         fxs-gs          up
        ds1-0/8@nytnk116       0:1         fxs-gs          up
        ds1-0/9@nytnk116       0:1         fxs-gs          up
        ds1-0/10@nytnk116      0:1         fxs-gs          up
        ds1-0/11@nytnk116      0:1         fxs-gs          up
        ds1-0/12@nytnk116      0:1         fxs-gs          up
        ds1-0/13@nytnk116      0:1         fxs-gs          up
        ds1-0/14@nytnk116      0:1         fxs-gs          up
        ds1-0/15@nytnk116      0:1         fxs-gs          up
        ds1-0/16@nytnk116      0:1         fxs-gs          up
        ds1-0/17@nytnk116      0:1         fxs-gs          up
        ds1-0/18@nytnk116      0:1         fxs-gs          up
        ds1-0/19@nytnk116      0:1         fxs-gs          up
        ds1-0/20@nytnk116      0:1         fxs-gs          up
        ds1-0/21@nytnk116      0:1         fxs-gs          up
        ds1-0/22@nytnk116      0:1         fxs-gs          up
        ds1-0/23@nytnk116      0:1         fxs-gs          up
        ds1-0/24@nytnk116      0:1         fxs-gs          up
```

continued below



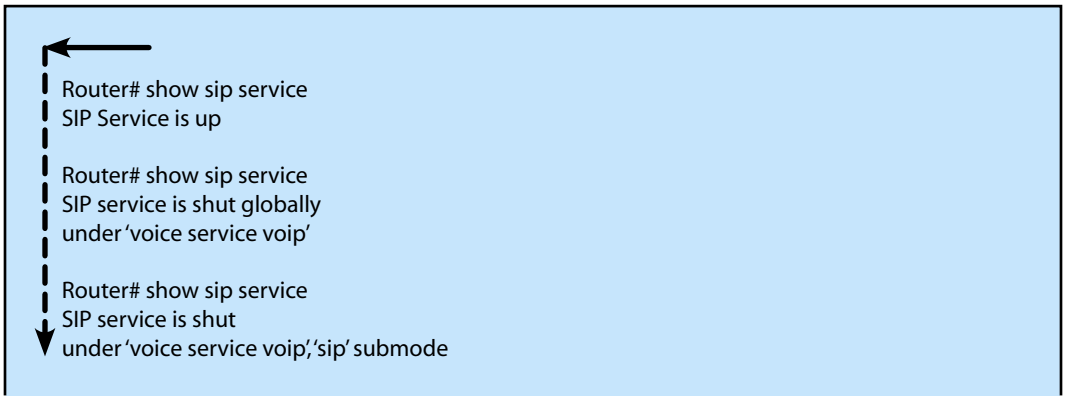
show proxy h323 calls

This command is used to display information about all active calls on the proxy.



show sip service

This command is used to display information about the SIP call service.



continued below

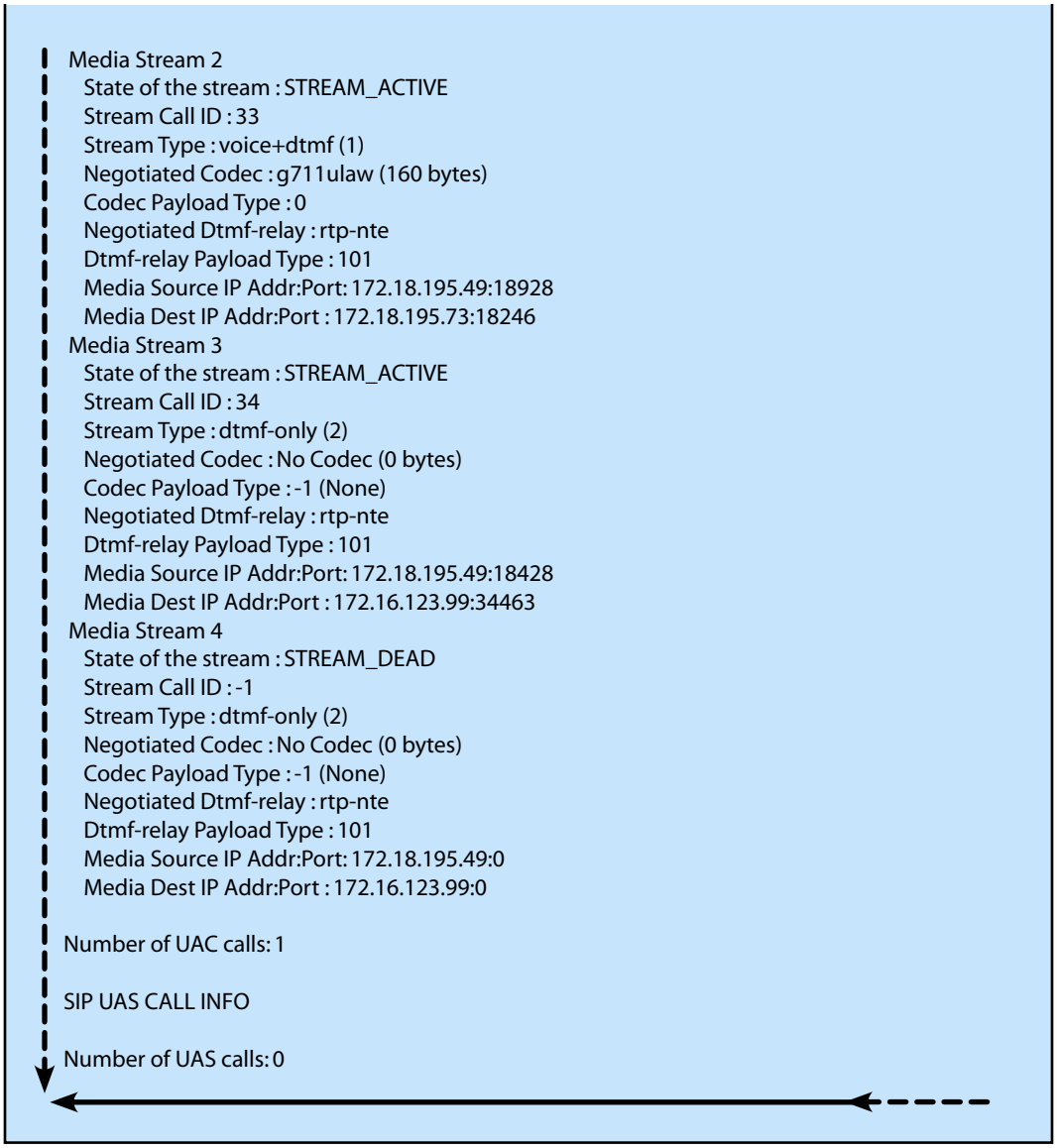
```
Router# show sip service
SIP service is forced shut globally
under 'voice service voip'

Router# show sip service
SIP service is forced shut
under 'voice service voip', 'sip' submode
```

show sip-ua calls

This command is used to display information about all active SIP UAC and UAS calls.

```
Router# show sip-ua calls

SIP UAC CALL INFO

Call 1
 SIP Call ID : 515205D4-20B711D6-8015FF77-1973C402@172.18.195.49
 State of the call : STATE_ACTIVE (6)
 Substate of the call : SUBSTATE_NONE (0)
 Calling Number : 5550200
 Called Number : 5551101
 Bit Flags : 0x12120030 0x220000
 Source IP Address (Sig ): 172.18.195.49
 Destn SIP Req Addr:Port : 172.18.207.18:5063
 Destn SIP Resp Addr:Port: 172.18.207.18:5063
 Destination Name : 172.18.207.18
 Number of Media Streams : 4
 Number of Active Streams: 3
 RTP Fork Object : 0x637C7B60
 Media Stream 1
   State of the stream : STREAM_ACTIVE
   Stream Call ID : 28
   Stream Type : voice-only (0)
   Negotiated Codec : g711ulaw (160 bytes)
   Codec Payload Type : 0
   Negotiated Dtmf-relay : inband-voice
   Dtmf-relay Payload Type : 0
   Media Source IP Addr:Port: 172.18.195.49:19444
   Media Dest IP Addr:Port : 172.18.193.190:16890
```

continued below


Media Stream 2
State of the stream : STREAM_ACTIVE
Stream Call ID : 33
Stream Type : voice+dtmf (1)
Negotiated Codec : g711ulaw (160 bytes)
Codec Payload Type : 0
Negotiated Dtmf-relay : rtp-nte
Dtmf-relay Payload Type : 101
Media Source IP Addr:Port: 172.18.195.49:18928
Media Dest IP Addr:Port : 172.18.195.73:18246
Media Stream 3
State of the stream : STREAM_ACTIVE
Stream Call ID : 34
Stream Type : dtmf-only (2)
Negotiated Codec : No Codec (0 bytes)
Codec Payload Type : -1 (None)
Negotiated Dtmf-relay : rtp-nte
Dtmf-relay Payload Type : 101
Media Source IP Addr:Port: 172.18.195.49:18428
Media Dest IP Addr:Port : 172.16.123.99:34463
Media Stream 4
State of the stream : STREAM_DEAD
Stream Call ID : -1
Stream Type : dtmf-only (2)
Negotiated Codec : No Codec (0 bytes)
Codec Payload Type : -1 (None)
Negotiated Dtmf-relay : rtp-nte
Dtmf-relay Payload Type : 101
Media Source IP Addr:Port: 172.18.195.49:0
Media Dest IP Addr:Port : 172.16.123.99:0
Number of UAC calls: 1
SIP UAS CALL INFO
Number of UAS calls: 0

show sip-ua status

This command is used to display information about the status of the SIP UA.

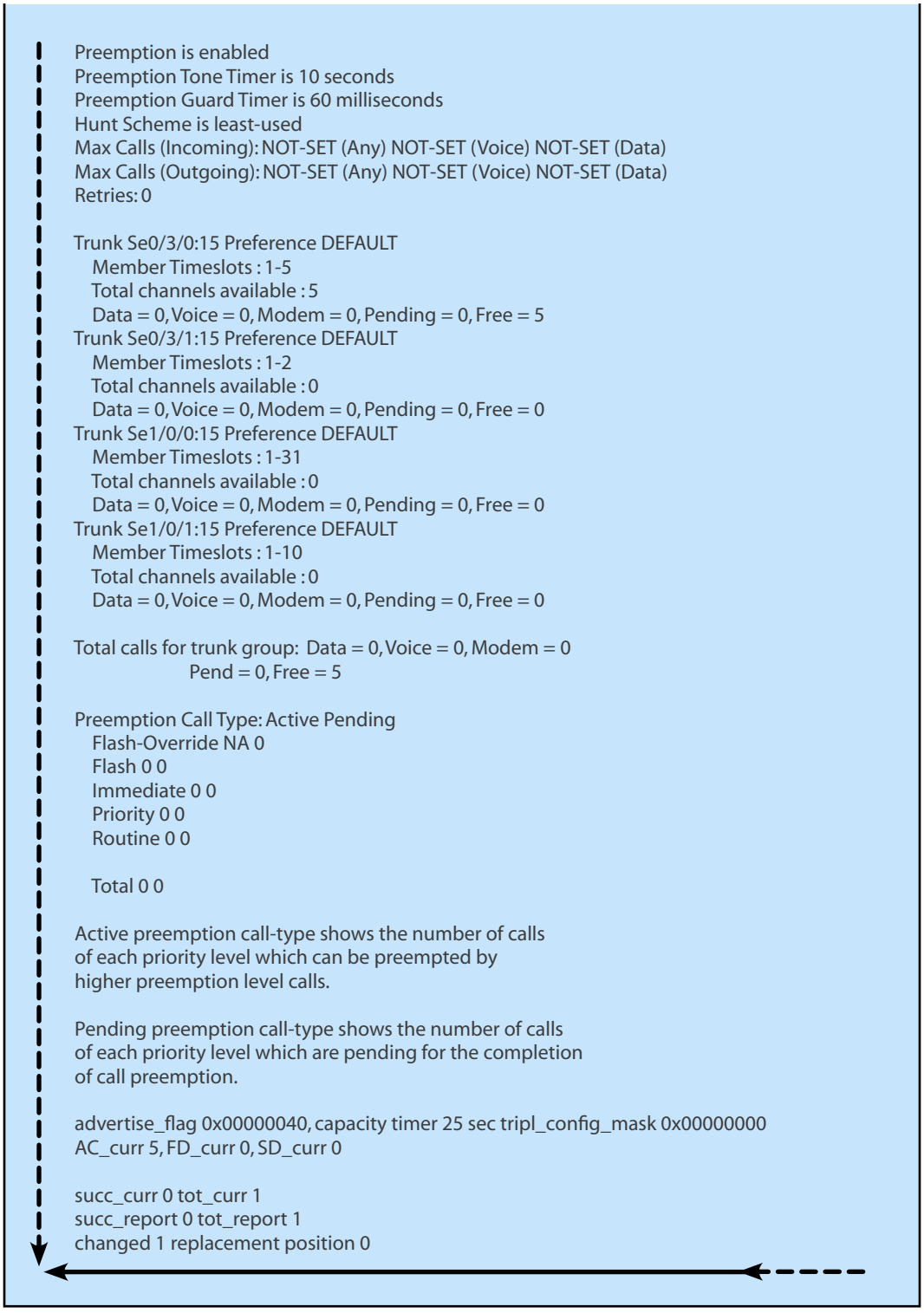
```
Router# show sip-ua status

SIP User Agent Status
SIP User Agent for UDP : ENABLED
SIP User Agent for TCP : ENABLED
SIP User Agent bind status(signaling): DISABLED
SIP User Agent bind status(media): DISABLED
SIP early-media for 180 responses with SDP: ENABLED
SIP max-forwards : 70
SIP DNS SRV version: 2 (rfc 2782)
NAT Settings for the SIP-UA
Role in SDP: NONE
Check media source packets: DISABLED
Maximum duration for a telephone-event in NOTIFYs: 2000 ms
SIP support for ISDN SUSPEND/RESUME: ENABLED
Redirection (3xx) message handling: ENABLED
Reason Header will override Response/Request Codes: ENABLED

 SDP application configuration:
 Version line (v=) required
 Owner line (o=) required
 Session name line (s=) required
 Timespec line (t=) required
 Media supported: audio image
 Network types supported: IN
 Address types supported: IP4
 Transport types supported: RTP/AVP udptl
```

show trunk group

This command is used to display information about configured trunk groups.

```
Router# show trunk group 1
Trunk group: 1

  Description:
  trunk group label: 1

  Translation profile (Incoming):
  Translation profile (Outgoing):
```

continued below



```
Preemption is enabled
Preemption Tone Timer is 10 seconds
Preemption Guard Timer is 60 milliseconds
Hunt Scheme is least-used
Max Calls (Incoming): NOT-SET (Any) NOT-SET (Voice) NOT-SET (Data)
Max Calls (Outgoing): NOT-SET (Any) NOT-SET (Voice) NOT-SET (Data)
Retries: 0

Trunk Se0/3/0:15 Preference DEFAULT
   Member Timeslots : 1-5
   Total channels available : 5
   Data = 0, Voice = 0, Modem = 0, Pending = 0, Free = 5
Trunk Se0/3/1:15 Preference DEFAULT
   Member Timeslots : 1-2
   Total channels available : 0
   Data = 0, Voice = 0, Modem = 0, Pending = 0, Free = 0
Trunk Se1/0/0:15 Preference DEFAULT
   Member Timeslots : 1-31
   Total channels available : 0
   Data = 0, Voice = 0, Modem = 0, Pending = 0, Free = 0
Trunk Se1/0/1:15 Preference DEFAULT
   Member Timeslots : 1-10
   Total channels available : 0
   Data = 0, Voice = 0, Modem = 0, Pending = 0, Free = 0

Total calls for trunk group:  Data = 0, Voice = 0, Modem = 0
                    Pend = 0, Free = 5

Preemption Call Type: Active Pending
   Flash-Override NA 0
   Flash 0 0
   Immediate 0 0
   Priority 0 0
   Routine 0 0

   Total 0 0

Active preemption call-type shows the number of calls
of each priority level which can be preempted by
higher preemption level calls.

Pending preemption call-type shows the number of calls
of each priority level which are pending for the completion
of call preemption.

advertise_flag 0x00000040, capacity timer 25 sec tripl_config_mask 0x00000000
AC_curr 5, FD_curr 0, SD_curr 0

succ_curr 0 tot_curr 1
succ_report 0 tot_report 1
changed 1 replacement position 0
```

show voice call summary

This command is used to display summary information about call status on voice ports.

```
Router# show voice call summary

PORT     CODEC     VAD   VTSP STATE      VPM STATE
======= ======== ===   ==========      ====================
0:17.18                                  *shutdown*
0:18.19   g729ar8    n     S_CONNECT       FXOLS_OFFHOOK
0:19.20                                  FXOLS_ONHOOK
0:20.21                                  FXOLS_ONHOOK
0:21.22                                  FXOLS_ONHOOK
0:22.23                                  FXOLS_ONHOOK
0:23.24                                  EM_ONHOOK
1/1                                      FXSLS_ONHOOK
1/2                                      FXSLS_ONHOOK
1/3                                      EM_ONHOOK
1/4                                      EM_ONHOOK
1/5                                      FXOLS_ONHOOK
1/6       g729ar8    n     S_CONNECT       FXOLS_CONNECT
```

show voice port

This command is used to display information about configured voice ports.

```
Router# show voice port 1/0/1

receEive and transMit Slot is 1, Sub-unit is 0, Port is 1
  Type of VoicePort is E&M
  Operation State is DORMANT
  Administrative State is UP
  No Interface Down Failure
  Description is not set
  Noise Regeneration is enabled
  Non Linear Processing is enabled
  Music On Hold Threshold is Set to -38 dBm
  In Gain is Set to 0 dB
  Out Attenuation is Set to 0 dB
  Echo Cancellation is enabled
  Echo Cancel Coverage is set to 8 ms
  Connection Mode is normal
  Connection Number is not set
  Initial Time Out is set to 10 s
  Interdigit Time Out is set to 10 s
  Region Tone is set for US
```

show voip rtp connections

This command is used to display information about RTP named event packets.

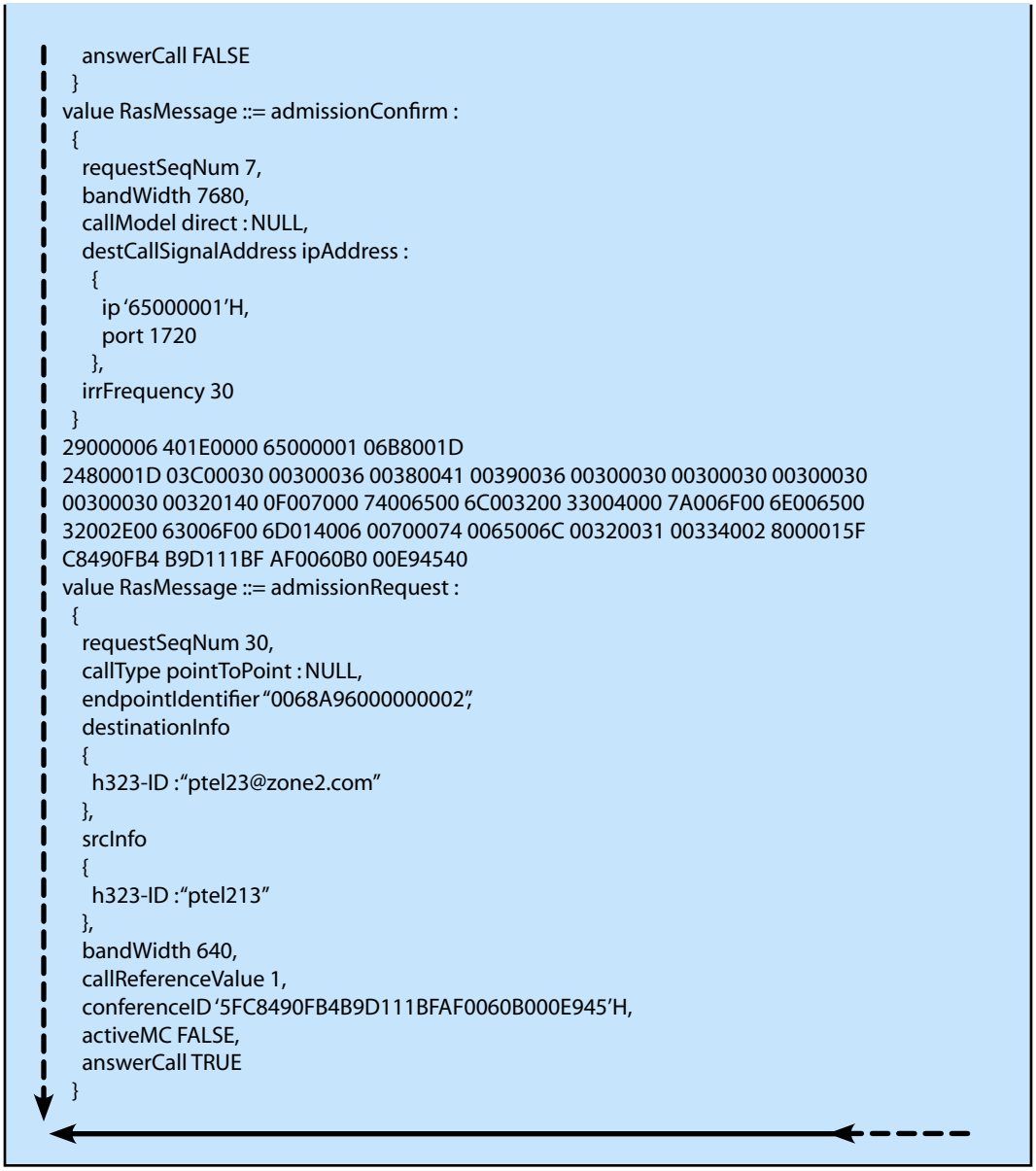
```
Router# show voip rtp connections

VoIP RTP active connections :
No.       CallId     dstCallId   LocalRTP   RmtRTP   LocalIP         RemoteIP
1         21         24          16996      18174    10.4.204.37     10.4.204.24
Found 1 active RTP connections
```

debug h225 asn1

This command is used to display information about the ASN1 contents of RAS and Q.931 messages.

```
Router# debug h225 asn1

H.225 ASN1 Messages debugging is on

Router#
24800006 03C00030 00300036 00380041 00450037 00430030 00300030 00300030
00300030 00310140 0F007000 74006500 6C003200 33004000 7A006F00 6E006500
32002E00 63006F00 6D020180 AAAA4006 00700074 0065006C 00320031 0033401E
0000015F C8490FB4 B9D111BF AF0060B0 00E94500

value RasMessage ::= admissionRequest :
 {
  requestSeqNum 7,
  callType pointToPoint : NULL,
  endpointIdentifier "0068AE7C00000001",
  destinationInfo
  {
   h323-ID :"ptel23@zone2.com"
  },
  srcInfo
  {
   e164 :"7777",
   h323-ID :"ptel213"
  },
  bandWidth 7680,
  callReferenceValue 1,
  conferenceID '5FC8490FB4B9D111BFAF0060B000E945'H,
  activeMC FALSE,
```

continued below



```
    answerCall FALSE
  }
value RasMessage ::= admissionConfirm :
  {
    requestSeqNum 7,
    bandWidth 7680,
    callModel direct : NULL,
    destCallSignalAddress ipAddress :
     {
       ip '65000001'H,
       port 1720
     },
    irrFrequency 30
  }
29000006 401E0000 65000001 06B8001D
2480001D 03C00030 00300036 00380041 00390036 00300030 00300030 00300030
00300030 00320140 0F007000 74006500 6C003200 33004000 7A006F00 6E006500
32002E00 63006F00 6D014006 00700074 0065006C 00320031 00334002 8000015F
C8490FB4 B9D111BF AF0060B0 00E94540
value RasMessage ::= admissionRequest :
  {
    requestSeqNum 30,
    callType pointToPoint : NULL,
    endpointIdentifier "0068A96000000002",
    destinationInfo
    {
      h323-ID :"ptel23@zone2.com"
    },
    srcInfo
    {
      h323-ID :"ptel213"
    },
    bandWidth 640,
    callReferenceValue 1,
    conferenceID '5FC8490FB4B9D111BFAF0060B000E945'H,
    activeMC FALSE,
    answerCall TRUE
  }
```

debug h225 events

This command is used to display information about the Q.931 events.

```
Router# debug h225 events

H.225 Event Messages debugging is on

Router#

H225Lib::h225TAccept: TCP connection accepted from 50.0.0.12:1701 on
socket [2]
    H225Lib::h225TAccept: Q.931 Call State is initialized to be [Null].
Hex representation of the received TPKT
0300007408020001050404889886A56C0580373737377E005B0500B0060008914A000101400
6007000740065006C00320031003302000140 0F007000740065006C003200330040007A006F0
06E00650032002E0063006F006D004EC8490FB4B9D111BFAF0060B000E945000C07003200000
C06B8
    H225Lib::h225RecvData: Q.931 SETUP received from socket [2]
    H225Lib::h225RecvData: State changed to [Call Present].
Hex representation of the CALL PROCEEDING TPKT to send.
0300001B08028001027E000F050100060008914A00010880012800
    H225Lib::h225CallProcRequest: Q.931 CALL PROCEEDING sent from socket
[2]. Call state remains unchanged (Q.931 FSM simplified for H.225.0)
    H225Lib::h225TConn: connect in progress on socket [4]
    H225Lib::h225TConn: Q.931 Call State is initialized to be [Null].
Hex representation of the SETUP TPKT to send.
030000A6080200840504048898 8CA56C0591373737377E008D0500B8060008914A000101400
6007000740065006C00320031003328 00B5000012400128000 1400F007000740065006C00320
0330040007A006F006E00650032002E0063006F006D006600000106B8004EC8490FB4B9D111B
FAF0060B000E945000E07006500000106B822400F007000740065006C003200330040007A006
F006E00650032002E0063006F006D
    H225Lib::h225SetupRequest: Q.931 SETUP sent from socket [4]
    H225Lib::h225SetupRequest: Q.931 Call State changed to [Call Initiated].
Hex representation of the received TPKT
0300001B08028084027E000F050100060008914A00010880012800
    H225Lib::h225RecvData: Q.931 CALL PROCEEDING received from socket [4]
Hex representation of the received TPKT
0300001808028084017E000C050300060008914A00010000
    H225Lib::h225RecvData: Q.931 ALERTING received from socket [4]
    H225Lib::h225RecvData: Q.931 Call State changed to [Call Delivered].
Hex representation of the ALERTING TPKT to send.
0300001808028001017E000C050300060008914A00010000
    H225Lib::h225AlertRequest: Q.931 ALERTING sent from socket [2]. Call
state changed to [Call Received].
Hex representation of the received TPKT
0300003708028084070403 88C0A57E00260502400600089 14A000100660000012AFF0880012
8004EC8490FB4B9D111BFAF0060B000E945
    H225Lib::h225RecvData: Q.931 CONNECT received from socket [4]
    H225Lib::h225RecvData: Q.931 Call State changed to [Active].
Hex representation of the CONNECT TPKT to send.
0300003808028001070404889886A57E00260502400600089 14A000100650000012AFC08800
128004EC8490FB4B9D111BFAF0060B000E945
    H225Lib::h225SetupResponse: Q.931 CONNECT sent from socket [2]
    H225Lib::h225SetupResponse: Q.931 Call State changed to [Active].
```

debug h245 asn1

This command is used to display information about the ASN1 contents of the H.245 messages.

```
*Mar  1 00:34:17.749: H245 MSC OUTGOING PDU ::=
value MultimediaSystemControlMessage ::= indication : userInput : alphanumeric :"1"

*Mar  1 00:34:17.749: H245 MSC OUTGOING ENCODE BUFFER::= 6D 400131
*Mar  1 00:34:17.753:
*Mar  1 00:34:18.350: H245 MSC OUTGOING PDU ::=
value MultimediaSystemControlMessage ::= indication : userInput : alphanumeric :"2"

*Mar  1 00:34:18.350: H245 MSC OUTGOING ENCODE BUFFER::= 6D 400132
*Mar  1 00:34:18.350:
*Mar  1 00:34:18.838: H245 MSC OUTGOING PDU ::=
value MultimediaSystemControlMessage ::= indication : userInput : alphanumeric :"3"

*Mar  1 00:34:18.838: H245 MSC OUTGOING ENCODE BUFFER::= 6D 400133
```

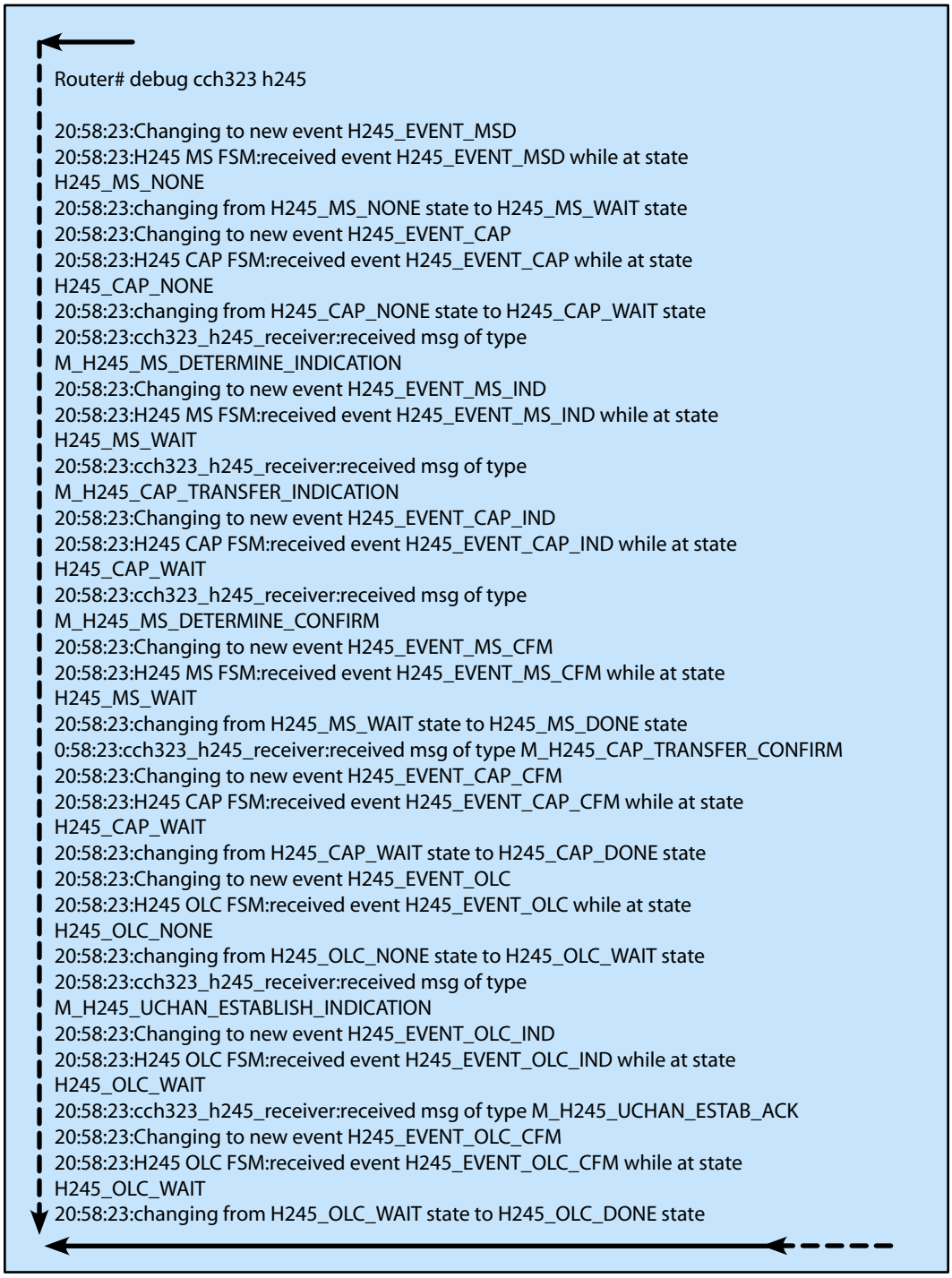
debug cch323 h225

This command is used to display information about the trace information about the H.225 state machine.

```
Router# debug cch323 h225

20:59:17:Set new event H225_EVENT_SETUP
20:59:17:H225 FSM:received event H225_EVENT_SETUP while at state H225_IDLE
20:59:17:Changing from H225_IDLE state to H225_SETUP state
20:59:17:cch323_h225_receiver:received msg of type SETUPCFM_CHOSEN
20:59:17:H225 FSM:received event H225_EVENT_SETUP_CFM_IND while at state
H225_SETUP
20:59:17:Changing from H225_SETUP state to H225_ACTIVE state
20:59:17:Set new event H225_EVENT_H245_SUCCESS
20:59:17:H225 FSM:received event H225_EVENT_H245_SUCCESS while at state
H225_ACTIVE
20:59:20:Set new event H225_EVENT_RELEASE
20:59:20:H225 FSM:received event H225_EVENT_RELEASE while at state
H225_ACTIVE
20:59:20:Changing from H225_ACTIVE state to H225_WAIT_FOR_DRQ state
20:59:20:Set new event H225_EVENT_RAS_SUCCESS
20:59:20:H225 FSM:received event H225_EVENT_RAS_SUCCESS while at state
H225_WAIT_FOR_DRQ
20:59:20:Changing from H225_WAIT_FOR_DRQ state to H225_IDLE state
```

debug cch323 h245

This command is used to display information about the trace information about the H.245 state machine.



```
Router# debug cch323 h245

20:58:23:Changing to new event H245_EVENT_MSD
20:58:23:H245 MS FSM:received event H245_EVENT_MSD while at state
H245_MS_NONE
20:58:23:changing from H245_MS_NONE state to H245_MS_WAIT state
20:58:23:Changing to new event H245_EVENT_CAP
20:58:23:H245 CAP FSM:received event H245_EVENT_CAP while at state
H245_CAP_NONE
20:58:23:changing from H245_CAP_NONE state to H245_CAP_WAIT state
20:58:23:cch323_h245_receiver:received msg of type
M_H245_MS_DETERMINE_INDICATION
20:58:23:Changing to new event H245_EVENT_MS_IND
20:58:23:H245 MS FSM:received event H245_EVENT_MS_IND while at state
H245_MS_WAIT
20:58:23:cch323_h245_receiver:received msg of type
M_H245_CAP_TRANSFER_INDICATION
20:58:23:Changing to new event H245_EVENT_CAP_IND
20:58:23:H245 CAP FSM:received event H245_EVENT_CAP_IND while at state
H245_CAP_WAIT
20:58:23:cch323_h245_receiver:received msg of type
M_H245_MS_DETERMINE_CONFIRM
20:58:23:Changing to new event H245_EVENT_MS_CFM
20:58:23:H245 MS FSM:received event H245_EVENT_MS_CFM while at state
H245_MS_WAIT
20:58:23:changing from H245_MS_WAIT state to H245_MS_DONE state
0:58:23:cch323_h245_receiver:received msg of type M_H245_CAP_TRANSFER_CONFIRM
20:58:23:Changing to new event H245_EVENT_CAP_CFM
20:58:23:H245 CAP FSM:received event H245_EVENT_CAP_CFM while at state
H245_CAP_WAIT
20:58:23:changing from H245_CAP_WAIT state to H245_CAP_DONE state
20:58:23:Changing to new event H245_EVENT_OLC
20:58:23:H245 OLC FSM:received event H245_EVENT_OLC while at state
H245_OLC_NONE
20:58:23:changing from H245_OLC_NONE state to H245_OLC_WAIT state
20:58:23:cch323_h245_receiver:received msg of type
M_H245_UCHAN_ESTABLISH_INDICATION
20:58:23:Changing to new event H245_EVENT_OLC_IND
20:58:23:H245 OLC FSM:received event H245_EVENT_OLC_IND while at state
H245_OLC_WAIT
20:58:23:cch323_h245_receiver:received msg of type M_H245_UCHAN_ESTAB_ACK
20:58:23:Changing to new event H245_EVENT_OLC_CFM
20:58:23:H245 OLC FSM:received event H245_EVENT_OLC_CFM while at state
H245_OLC_WAIT
20:58:23:changing from H245_OLC_WAIT state to H245_OLC_DONE state
```

Domain 5 - Describe the Function and Interoperation of Gatekeepers within an IP Communications Network

Describe the Function and Types of Gatekeepers

H.323 gatekeeper is used in larger networks in order to centralize several of the VoIP functions. The gatekeepers do this by separating the VoIP network into zones. Zones are allowed to cross subnets and can manage multiple subnets. Each individual gatekeeper can manage up to 100 zones.

The main duties of the gatekeeper include address resolution, admission control, bandwidth management, zone management and call authorization. The gatekeeper resolves H.323 ID's and E.164 telephone numbers to the IP address of the destination gatekeeper. The admission control function of the gatekeeper is used when a gateway gets a call request, at this point the gateway contacts its gatekeeper to ensure that the call is allowed to be placed. The bandwidth management feature of the gatekeeper is used to maintain the maximum bandwidth used by calls both within the zone and between zones. If the bandwidth required to place a new call is insufficient then the call is rejected. The zone management feature of the gatekeeper keeps track of all registered gateways in the zones it manages. All gateways register with the gatekeeper when they come online. The call authorization functionality of the gatekeeper allows control of the access to and from specific gateways and endpoints, on Cisco equipment this is typically done with AAA.

Describe the Interoperation of Devices with a Gatekeeper

Devices in a H.323 network communicate in two ways, either directly (if both devices are H.323 terminal devices or are connected through a H.323 gateway) or through a gatekeeper. IF using a gatekeeper the devices (terminals and gateways) are also responsible for registering with the gatekeeper. This communication is done through a number of different Registration, Admission and Status (RAS) messages. These communications are called signaling and are detailed in the next section. Once the gateways have communicated with the gatekeeper and the destination gateway is found then the two gateways are allowed to communicate directly for the length of the call until it is disconnected. The request for call termination is also relayed through the gatekeeper.

Describe Gatekeeper Signaling

As stated above in the previous section there is a number of different RAS messages which are defined for H.323. These messages are used for a variety of purposes from registration to call admission requests. The following table shows the various important RAS messages used between the endpoint device, gateway and gatekeeper.

Terminal and Gateway Discovery Messages	
GRQ	Gatekeeper Request
GCF	Gatekeeper Confirmation
GRJ	Gatekeeper Reject

table continued

Terminal and Gateway Registration Messages	
RRQ	Registration Request
RCF	Registration Confirmation
RRJ	Registration Reject
Terminal and Gateway Unregistration Messages	
URQ	Unregistration Request
UCF	Unregistration Confirmation
URJ	Unregistration Reject
Terminal to Gatekeeper Admission Messages	
ARQ	Admission Request
ACF	Admission Confirmation
ARJ	Admission Reject
Terminal to Gatekeeper Bandwidth Change Messages	
BRQ	Bandwidth Request
BCF	Bandwidth Confirmation
BRJ	Bandwidth Reject
Location Request Messages	
LRQ	Location Request
LCF	Location Confirmation
LRJ	Location Reject
Disengage Request Messages	
DRQ	Disengage Request
DCF	Disengage Confirmation
DRJ	Disengage Reject
Status Request Messages	
IRQ	Information Request

table continued

IRR	Information Request Response
IACK	Information Request Acknowledge
INAK	Information Request Negative Acknowledge
Gateway Resource Availability Messages	
RAI	Resource Available Indication
RAC	Resource Available Confirmation

Table 6 - Common RAS Messages

Describe Dynamic Zone Prefix Registration with a Gatekeeper

When H.323 Version 4 was introduced one of the features that was introduced was Dynamic Zone Prefix registration. What this does is eliminate much of the configuration needed on the Gatekeeper. Previous to this feature all zone prefixes for all gateways under a gatekeeper needed to be statically configured. This of course was very time consuming and could lead to problems with ongoing problems as the configuration needed to be completed on the gateway and on the gatekeeper. When using this feature, all destination patterns configured on the POTS dial peers on the gateway will be dynamically registered with the gatekeeper. The gatekeeper then uses this information for call routing.

Describe Gatekeeper Redundancy

There are four main types of gatekeeper redundancy provided on Cisco equipment: redundant VoIP dial peer configuration, Hot Standby Routing Protocol (HSRP), alternate gatekeeper and gatekeeper clustering.

When redundant VoIP dial peers are used the gateways are primarily configured to communicate with the gatekeeper. In the case that the gatekeeper goes unreachable the gateway will failover to its lower priority VoIP dial peers. This type of configuration will work fine in small networks where VoIP dial peers can be adequately configured for the network but with larger networks this type of configuration becomes very inefficient because the gateway must be configured with VoIP dial peers which service the whole network.

HSRP can be used to have a separate redundant piece of equipment which backs up the primary equipment. This backup is only used when the primary fails and does not actively provide any service unless this happens. No call state information is maintained between primary and redundant equipment.

The introduction of H.323 Version 2 included a feature called Alternate gatekeeper. When this feature is used multiple gatekeepers can be configured to control one zone. When a gateway's primary gatekeeper goes down it fails over to one of its alternate gatekeepers. No Call state information is maintained between alternate gatekeepers.

A Cisco proprietary redundancy feature includes gatekeeper clustering. Gatekeeper clustering uses the Gatekeeper Update Protocol (GUP) to speak between clustered gatekeepers. With gatekeeper clustering call state information is maintained between gatekeepers and load balancing is possible between a cluster members. Gatekeeper clustering also includes the ability to provide rapid failover to other gatekeepers should a failure occur.

Domain 6 - Implement a Gatekeeper

Configure Devices to Register with a Gatekeeper

In order for a gateway to work with a gatekeeper it must first register itself and its prefixes must be configured either on the gatekeeper or must be registered through dynamic prefix registration by the gateway. By default on current IOS versions, dynamic prefix registration is enabled.

The first thing that must be done to configure the gateway is specify an H.323 interface. To specify an H.323 interface the **h323-gateway voip voip interface** command is used while in interface configuration mode. The second thing that is configured is the location of the gatekeeper for the gateway to use. This task is done using the **h323-gateway voip id** *gatekeeper-id* **{ipaddr** *ip-address* | **multicast}** command while entered in interface configuration mode. This command can either be entered statically with a specific IP address or the local gatekeeper can be discovered using multicast. The third optional thing that is configured is the H.323 ID of the gateway; this is configured using the **h323-gateway voip h323-id** *interface-id* command while in interface configuration mode. After these commands are issued gateway service on the equipment must be enabled using the **gateway** command while in global configuration mode.

It is also very important for a default technical prefix to be configured on the gateway and on the gatekeeper. In order to configure the default technical prefix on the gateway the **h323-gateway voip tech-prefix** *prefix* command is used while in interface configuration mode.

Once the gateway has been configured it is required for the gateway to direct all remote traffic to the gatekeeper. The command to do this is done with the **session target ras** command while in dial-peer voice configuration mode on the VoIP dial-peer.

Configure Gatekeeper to Provide Dial-Plan Resolution

The first thing that must be configured on the gatekeeper is to enable the gatekeeper service. To enable the gatekeeper services use the gatekeeper command while in global configuration mode. Once this is done it is required for all local zones being services by the gatekeeper to be configured. To configure these zones use the zone local *zone-name domain ip-address* command while in gatekeeper configuration mode. Once all the zone local commands are entered the gatekeeper services must be enabled with the no shutdown command while in gatekeeper configuration mode.

The default technical prefix is configured on the gatekeeper using the **gw-type-prefix** *prefix* **default-technology** command.

Prefixes can be configured on the gatekeeper in two main ways, with the **zone prefix** *zonename prefix* command. Prefixes can be dynamically registered by the gateway with a current IOS version. This can be enabled by configuring the gatekeeper with the **rrq dynamic prefixes-accept** command while in interface configuration mode and configuring the gateway with the **ras rrq dynamic prefixes** command while in the H.323 service configuration mode.

Configure Gatekeeper to Provide Call Admission Control

Call Admission Control (CAC) on the gatekeeper can be configured in a couple of different ways: Interzone, Total, Session and Remote. The Interzone option enables the configuration of the max amount of bandwidth from the zone to all other zones. The total option enables the configuration of the max amount of bandwidth allowed for all calls in a zone. The Session option enables the configuration of the max amount of bandwidth allowed for a session in a zone. The Remote option enables the configuration of the max amount of bandwidth allowed to all remote zones managed by other gatekeepers.

CAC is configured on the gatekeepers using one of two commands: **bandwidth {interzone | total | session} zone** *zonename max-bandwidth* and **bandwidth remote** max-bandwidth.

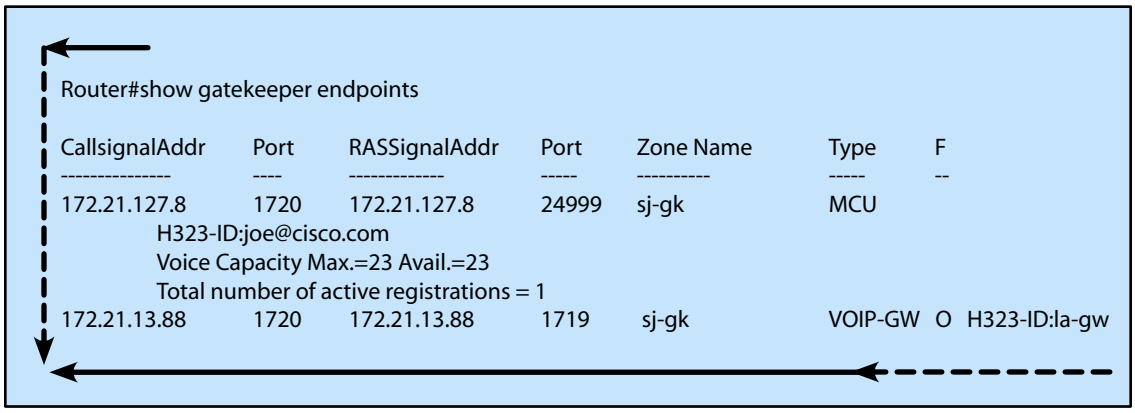
Verify Gatekeeper Operation

In order to verify that all of the gateways have correctly registered with the gatekeeper the show gatekeeper endpoints command can be used. If problems are found with registering gateways with the gatekeeper then the debug h225 asn1 command can be used, this command will display all discovery trace information between the gateways and the gatekeeper. If there are call routing problems which are found between the gatekeeper and the gateways then the debug gatekeeper main 5 command can be used. This command will enable the display of decision tree steps the gatekeeper takes when processing the ARQ RAS messages.

Troubleshooting

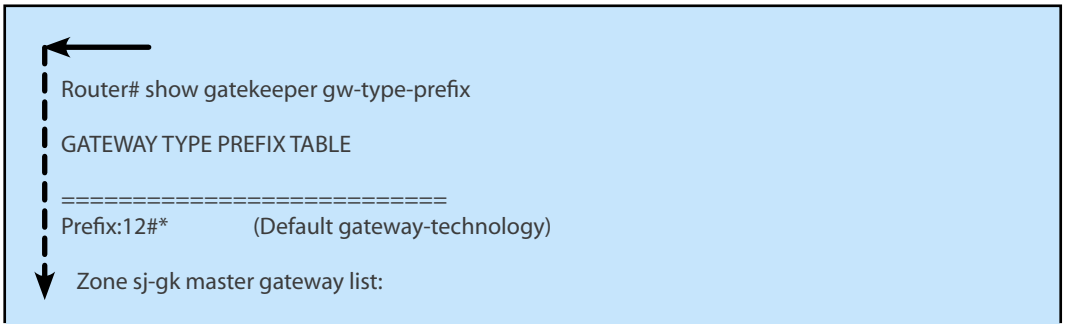
show gatekeeper endpoints

This command is used to display information about the registered endpoints with a gatekeeper.

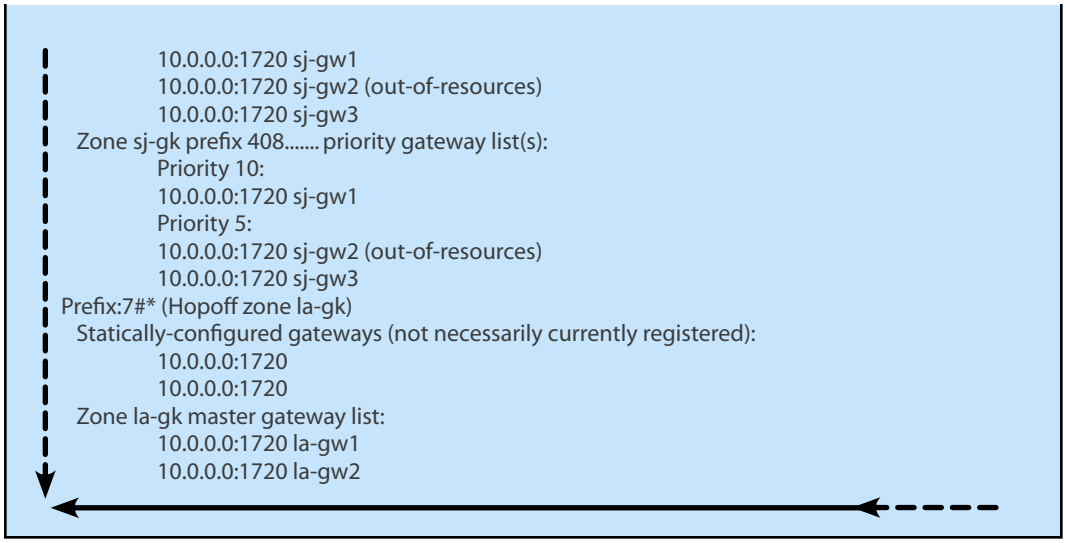


show gatekeeper gw-type-prefix

This command is used to display information about the gateway technology prefix table.

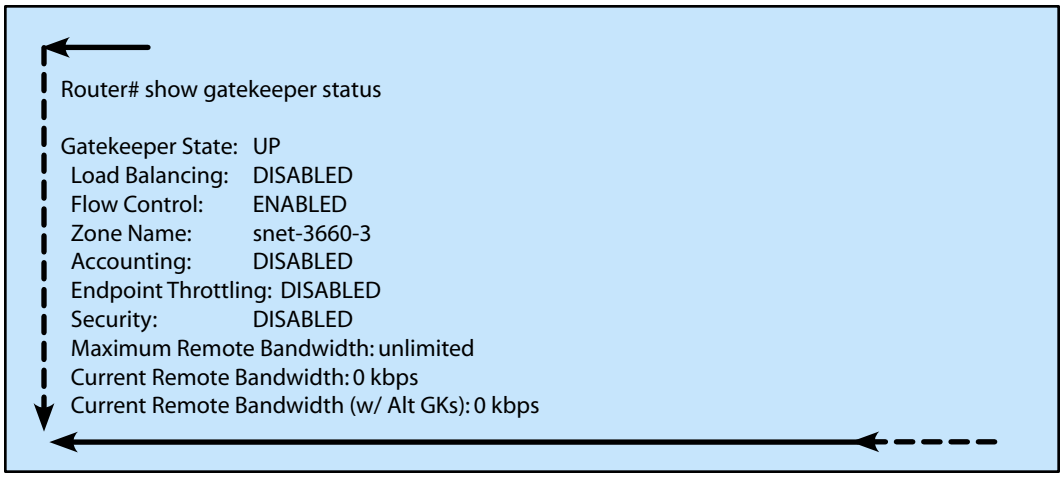


continued below



show gatekeeper status

This command is used to display information about the overall gatekeeper status.



show gatekeeper zone prefix

This command is used to display information about the registered gatekeeper zone prefix table.

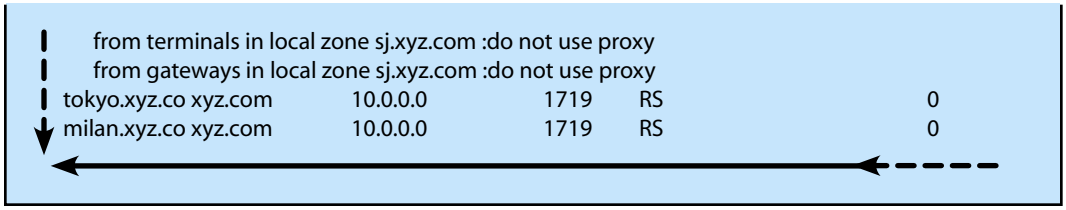
```
Router# show gatekeeper zone prefix
            ZONE PREFIX TABLE
            =================
GK-NAME                  E164-PREFIX
-------                  -----------
gk2                      408*
gk2                      5551001*
gk2                      5551002*
gk2                      5553020*
gk2                      5553020*
gk1                      555....
gk2                      719*
gk2                      919*
```

show gatekeeper zone status

This command is used to display information about the status of zones associated with a gatekeeper.

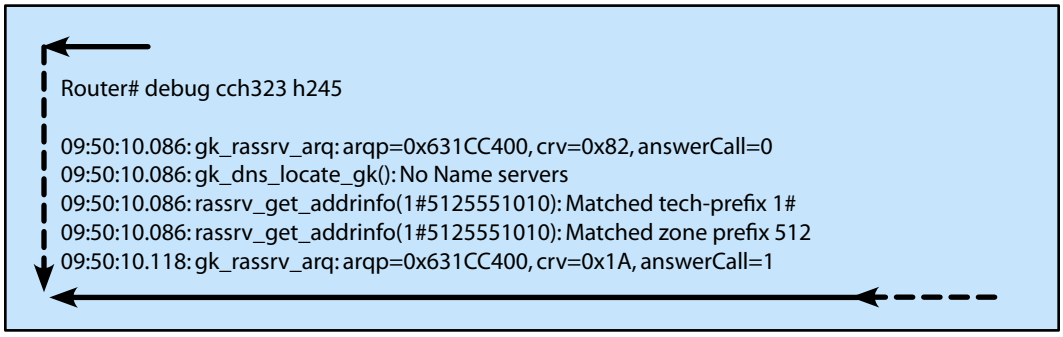
```
Router# show gatekeeper zone status

GATEKEEPER ZONES
================
GK name Domain Name      RAS Address      PORT    FLAGS   MAX-BW (kbps)   CUR-BW (kbps)
-------  -----------     -----------      ----    -----   ------          ------
sj.xyz.com xyz.com       10.0.0.0         1719    LS                      0
 SUBNET ATTRIBUTES :
  All Other Subnets :(Enabled)
 PROXY USAGE CONFIGURATION :
  inbound Calls from germany.xyz.com :
   to terminals in local zone sj.xyz.com :use proxy
   to gateways in local zone sj.xyz.com :do not use proxy
  Outbound Calls to germany.xyz.com
   from terminals in local zone germany.xyz.com :use proxy
   from gateways in local zone germany.xyz.com :do not use proxy
  Inbound Calls from all other zones :
   to terminals in local zone sj.xyz.com :use proxy
   to gateways in local zone sj.xyz.com :do not use proxy
  Outbound Calls to all other zones :
```

continued below



debug gatekeeper main 10

This command is used to display information about the gatekeeper prefix processing logic.



Domain 7 - Implement an IP-to-IP Gateway

Describe the IP-to-IP Gateway Features and Functionality

The purpose of the IP-to-IP gateway being created was to find a more elegant solution to connecting two providers VoIP networks together. Before these gateways were possible this connection was done via TDM circuits between both providers. At this demarcation point the gateways on both sides were required to decode the VoIP back onto a TDM circuit then have it recoded back into a VoIP codec again on the opposing provider's gateway. This act introduced a delay into the call and required that both providers dedicate DSP resources for this action.

In response for a better solution the IP-to-IP gateway was created, also called the Session Border Controller (SBC). It is the purpose of the SBC to act as a gateway between two separate providers' VoIP networks. Cisco's implementation of this is the Cisco Multiservice IP-to-IP Gateway (IPIPGW). When using an IOS version that supports IPIPGW a Cisco device can act as this SBC between two providers' VoIP networks. Specifically the IPIPGW has the ability to act as a gateway with SIP and H.323 and can be used only in the following directions: SIP-to-SIP, H.323-to-H.323, SIP-to-H.323 and H.323-to-SIP. This feature does require a very specific IOS version to be supported (INT VOICE/VIDEO, IPIPGW, and TMDIP GW).

There are two ways which a SBC can operate, in media flow-through mode or media flow-around mode. While in media flow-through mode both the signaling and the media traffic flow terminate and re-originate at the IPIPGW. While in media flow-around mode the signaling traffic flow is terminated and re-originated at the IPIPGW, the media traffic flow does not terminate at the IPIPGW. When using media flow-through mode the sender and destination are completely obscured from one another, this also provides a way to change the type of codec, QoS and DTMF relaying method.

Configure Gatekeeper to Support an IP-to-IP Gateway

By default, a Cisco device will allow you to configure TDM-to-VoIP and VoIP-to-TDM call legs but will not allow the VoIP-to-VoIP call legs required with an IP-to-IP gateway. In order to enable this functionality the allow-connections command is used while in voice service configuration mode. The allow-connections command is also used to configure the IPIPGW to go between different signaling protocols. There are four variations of the allow-connections command: allow-connections sip to sip, allow-connections sip to h323, allow-connections h323 to h323 and allow-connections h323 to sip.

In order to configure an IPIPGW into either media flow-through or flow-around mode the **media {flow-through | flow-around}** command is used while in voice service configuration mode.

Once the above commands are configured the IPIPGW is configured as any normal gateway but with multiple VoIP dial peers. In order to configure a specific VoIP signaling protocol the **session protocol** command is used while in dial-peer voice configuration mode. By default, the session protocol is set to H.323. In order to configure the dial-peer as a SIP dial-peer the **session protocol sipv2** command is used.

Configure IP-to-IP Gateway to Provide Address Hiding

As stated above if the media flow-through mode is used all traffic from both sides of a connection is terminated and re-originated at the IPIPGW. When this configuration is used the IP addresses of both endpoints is completely obscured from each other because the IPIPGW's IP address is used as the source and destination.

One caveat should be noted, when an IPIPGW is used between a SIP and an H.323 network each protocol must be bound to different interfaces with different IP addresses.

Configure IP-to-IP Gateway to Provide Protocol and Media Interworking

One of the most useful features which can be used on the configured dial-peers is the codec transparent command. When the codec transparent command is used, if the codec is the same on both networks no decoding and recoding is done on the IPIPGW.

Another thing that can also be changed on the IPIPGW is the DTMF relay type, each different network may potentially use different DTMF relay types and this can be changed on the IPIPGW.

Fax support is also supported using the same commands that are discussed in previous sections.

Configure IP-to-IP Gateway to Provide Call Admission Control

CAC is provided on IP-to-IP gateways in a number of ways but there are four main ways which are currently in use: Resource ReSerVation Protocol (RSVP) CAC, Max Connections based CAC and IP Call Capacity-Based CAC.

RSVP CAC requires a great amount of configuration to setup and its detailed configuration is outside the scope of this document. Max Connection based CAC is simple and is used to limit the max number of connections allowed over a specific dial-peer. To configure max connection based CAC use the **max-conn** *max-connections* command while in dial-peer voice configuration mode.

IP Call Capacity-based CAC is specific to H.323, it uses the carrier ID assigned to a specific dial-peer to configure the max amount of calls reserved to a specific carrier. IP Call Capacity can be configured using the **ip circuit carrier-id** *carrier-id* **reserved-calls** *num-reserve-calls* which is entered in while in H.323 configuration mode.

The fourth method of CAC is based on specific thresholds, the threshold which can be monitored include: CPU utilization, memory use and total calls. In order to configure threshold based CAC the **call threshold global** *trigger-name* **low** *percent* **high** *percent* command which is entered in global configuration mode.

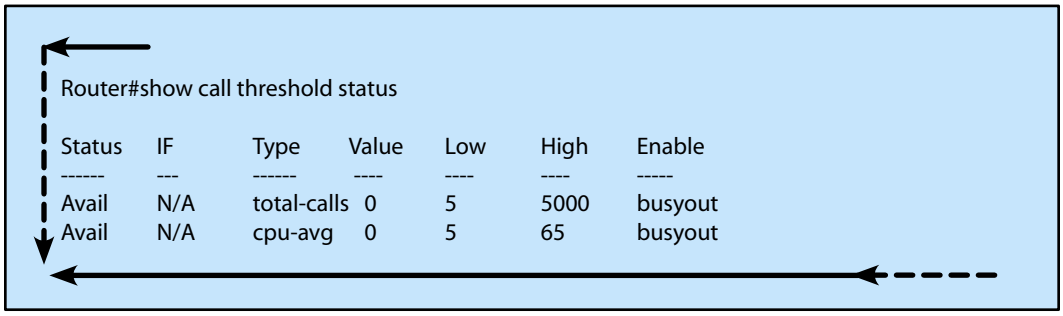
Verify IP-to-IP Gateway Implementations

There are no real show commands which are used to verify only IP-to-IP gateway functions. The use of the normal verification gateway commands can be used to show all RTP and dial-peer activity. There is however a debug command which can be used when troubleshooting is required, this is the debug voip ipipgw command.

Troubleshooting

show call threshold status

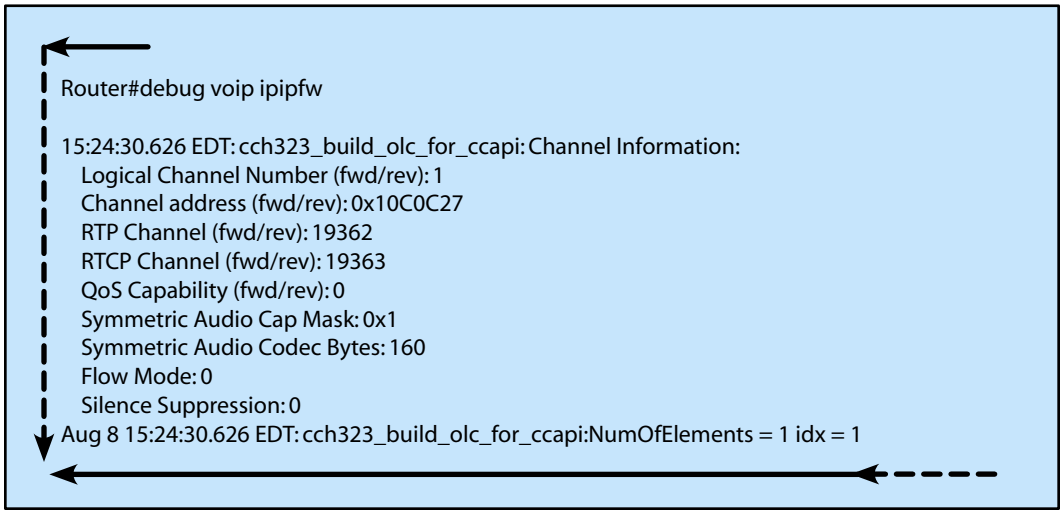
This command is used to display information about the configuration of all threshold triggers.



```
Router#show call threshold status

Status  IF   Type        Value  Low  High  Enable
------  ---  ------      ----   ---- ----  -----
Avail   N/A  total-calls  0     5    5000  busyout
Avail   N/A  cpu-avg     0     5    65    busyout
```

debug voip ipipgw

This command is used to display information about the events of the IPIPGW.



```
Router#debug voip ipipfw

15:24:30.626 EDT: cch323_build_olc_for_ccapi: Channel Information:
   Logical Channel Number (fwd/rev): 1
   Channel address (fwd/rev): 0x10C0C27
   RTP Channel (fwd/rev): 19362
   RTCP Channel (fwd/rev): 19363
   QoS Capability (fwd/rev): 0
   Symmetric Audio Cap Mask: 0x1
   Symmetric Audio Codec Bytes: 160
   Flow Mode: 0
   Silence Suppression: 0
Aug 8 15:24:30.626 EDT: cch323_build_olc_for_ccapi:NumOfElements = 1 idx = 1
```